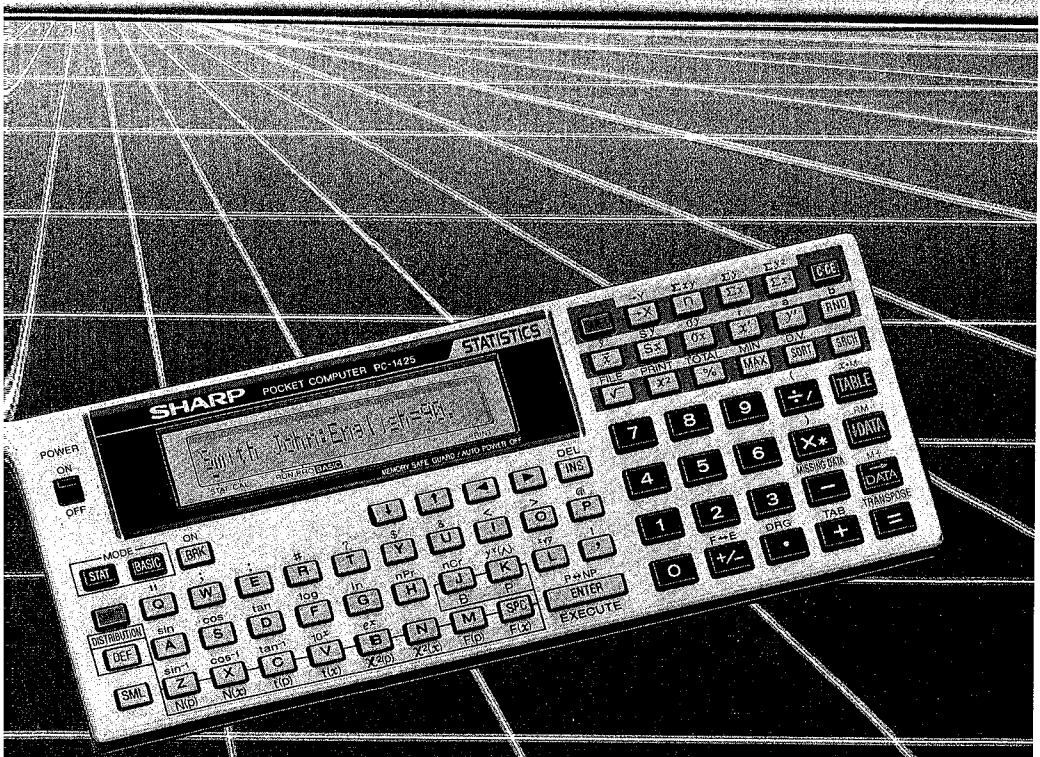
SHARP®

ASCHENCOMPUTER

MODELL

PC-1425

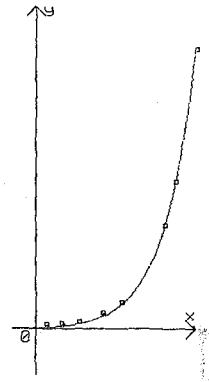
BEDIENUNGSANLEITUNG



FARBDRUCK

Exponentielle Regression und Plotten (Siehe Seite 294)

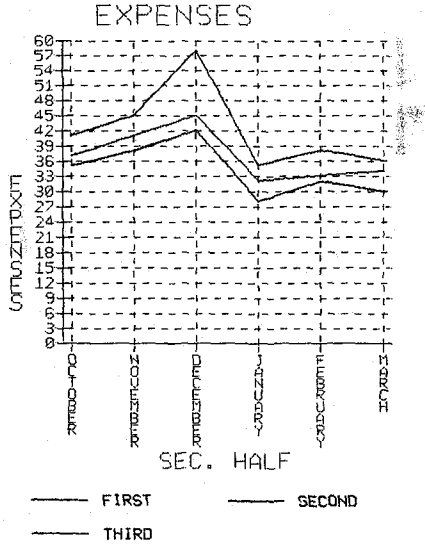
```
r=9.999942365E-01
a=4.960331916
b=2.03057723
```




```
* Estimation *
x=2          y=20.45265825
x=4          y=84.3312981
x=6          y=347.7185094
x=6.5        y=495.4930476
```

Erstellung einer kurvengrafik (Siehe Seite 312)
(Verbindung von Kalkulationslabellen-Daten mit BASIC)

	FIRST	SECOND	THIRD
OCTOBER	41	35	32
NOVEMBER	45	38	41
DECEMBER	58	42	45
JANUARY	35	28	32
FEBRUARY	38	32	33
MARCH	36	30	34



Maßstab um 55% verkleinert

PC-1425
BEDIENUNGSANLEITUNG

INHALT

EINLEITUNG 5

KAPITEL 1 VERWENDUNG DIESER BEDIENUNGSANLEITUNG 7
1.1 Tastaturabdeckung 8

KAPITEL 2 VORSTELLUNG DES COMPUTERS 9
2.1 Allgemeines 9
2.2 Bedienelemente 10
2.3 Die Anzeige 11
2.4 Wahl der Betriebsart 12
STAT- und CAL-Mode 13
RUN- und PRO-Mode 14
2.5 Betriebshinweise 14
2.6 Stromversorgung 16

KAPITEL 3 STATISTISCHE BERECHNUNGEN 23
3.1 Einschalten des Computers 23
3.2 Ausschalten des Computers 24
Automatisch 24
Manuell 24
3.3 Betriebsart für statistische Berechnungen 24
3.4 Kalkulationstabellen-Betriebsart 25
3.5 Editierfunktionen 34
3.6 Grundlegende Kalkulationstabellen-Funktionen 35
3.7 Sichern und Laden von Kalkulationstabellen 41
3.8 Statistische Berechnungen 43
3.9 Verwendung der Verteilungsfunktionen 49
3.9 Drucken der Kalkulationstabellen 52
3.10 Verwendung der Daten mit BASIC 54

KAPITEL 4 VERWENDUNG ALS WISSENSCHAFTLICHER RECHNER 57
4.1 Betriebshinweise 57
Zweite Funktionsebene der Tastatur, Merkmale, Editier- und Korrekturmöglichkeiten 57
Anzeige/Display-Format und Symbole 58
Grundeinstellung 59
Löschen der Eingabe oder einer Fehlermeldung 61
Auf-/Abrundung bzw. Festlegung der Nachkommastellen 61
Wissenschaftliche Schreibweise 62

4.2 Normale Berechnungen 63
Addition und Subtraktion 63
Multiplikation und Division 64
Potenz- und Wurzelfunktion 65
Prozentrechnung 65
Speicherrechnung (nur in der CAL-Betriebsart) 65
Vorrangordnung und Verwendung der "Klammern"-Tasten 66
4.3 Wissenschaftliche Berechnungen 69
Trigonometrische und inverse trigonometrische Funktionen (Arcusfunktionen) 69
Logarithmische Funktionen 70
Exponentialfunktionen 70
Fakultät 70
Permutation 71
Kombination 71
Zufallszahlen 71
4.4 Einsatz als BASIC-Rechner 72
Rechnen ohne Programmunterstützung (RUN-Mode) 72
Grundrechnungsarten 72
Rechengenauigkeit 74
Wissenschaftliche Schreibweise 75
Editier-/Korrekturmöglichkeit 76
Mathematische Funktionen/Klammerregeln 77
Hexadezimal-(Sedezimal-)Zahlen 78
Textausdrücke 78
Logische Vergleichsausdrücke 80

KAPITEL 5 BASIC — KONZEPT UND BEGRIFFE 83
5.1 Sprachelemente 83
Numerische Konstante 83
Textkonstante 83
Numerische Variable 83
Textvariable 84
Numerische Funktionen, Textfunktionen 84
Numerische Ausdrücke 85
Textausdrücke 85
Logische Vergleichsausdrücke 86
Logishe Ausdrücke 88
5.2 Variable 91
Standardvariable 92
Einfache Variablen 93
Indizierte Variable 93

Besonderheiten der Variablen A ... 94
Feldvariable ... 96
Löschen von Variablen ... 96
Numerische Feldvariable ... 97
Textfeldvariable ... 98
5.3 Arithmetische Funktionen ... 99
5.4 Textfunktionen ... 101

KAPITEL 6 PROGRAMMIEREN IN BASIC ... 103
6.1 BASIC-Übersicht ... 103
Programmerstellung ... 104
Programmaufbau ... 105
Eingabe eines Programms ... 105
Korrektur einer Zeile ... 107
Löschen oder Kopieren einer Zeile ... 108
Programmausführung ... 108
Fehlermeldungen/Fehlersuche ... 109

KAPITEL 7 PRAKTISCHE HINWEISE ... 111
7.1 Die DEF-Taste und etikettierte Programme ... 111
7.2 Tastatur-Schablone ... 112

KAPITEL 8 VERWENDUNG VON DRUCKER UND KASSETTEN-INTERFACE ... 113
8.1 Allgemeines ... 113
8.2 Verwendung als Drucker ... 114
Manueller Druckbetrieb (Protokollerstellung) ... 114
Programmgesteuerter Ausdruck ... 115
8.3 Verwendung des integrierten Kassetten-Interfaces ... 116
Auswahl des Kassettenrekorders ... 116
Anschluß des Kassettenrekorders an das CE-126P ... 117
8.4 Datenspeicherung auf Magnetband ... 117
Allgemeines ... 117
Speichern von Daten und Programmen ... 118
Überprüfen der Abspeicherung ... 119
Laden von Programmen oder Daten ... 119
8.5 Verwendung des CE-140P/-515P ... 120
8.6 Serielle E/A-Funktionen ... 121

KAPITEL 9 DIE RAM-KARTE 125
9.1 Austausch von RAM-Karten 126
9.2 Austausch von RAM-Karten der 1/2 Größe 126
9.3 Austausch von RAM-Karten der Standardgröße 129
9.4 Verwendung der RAM-Karten mit anderen Modellen 131

KAPITEL 10 BASIC REFERENZTEIL 133
10.1 Kommandos 136
10.2 Befehle 153
10.3 Funktionen 215
Verteilungsfunktionen 215
Pseudovariable 220
Numerische Funktionen 223
String-Funktionen 228
10.4 Ein-/Ausgabe-Kommandos 231
10.5 Text-Funktionskommandos 248

KAPITEL 11 FEHLERSUCHE 251

KAPITEL 12 RICHTIGE BEHANDLUNG DES COMPUTERS 255

ANHANG 257
ANHANG A Fehlermeldungen 257
ANHANG B ASCII-Code Tabelle 260
ANHANG C Formatieren der Datenausgabe 262
ANHANG D Bewertung von Ausdrücken und Operatoren-Vorrang 266
ANHANG E Tastenfunktionen 268
ANHANG F Signale des E/A-Anschlusses 276
ANHANG G Technische Daten 277
ANHANG H Die Benutzung von Programmen, die für andere PC-Modelle entwickelt wurden 279

PROGRAMMBEISPIELE 285
INDEX 329

EINLEITUNG

Willkommen in der Welt der SHARP-Besitzer!

Nur wenige Industriezweige der Welt können heute Schritt halten mit dem schnellen Wachstum und dem technischen Fortschritt im Bereich der Personal-Computer. Computer, die noch vor kurzer Zeit einen großen Raum gefüllt haben, deren Programmierung einen akademischen Abschluß erforderte und die Tausende von Dollars kosteten, passen heute in Ihre Handfläche, sind einfach zu programmieren und sind so preisgünstig, daß sie für jeden erschwinglich sind.
Ihr neuer SHARP-Computer bringt Ihnen die neuesten Fortschritte dieser technischen Revolution ins Haus. Er verfügt über viele praktische Funktionen:

* STATISTISCHE BERECHNUNGEN — Der Computer ist mit statistischen Funktionen ausgestattet, die auf Kalkulationstabellen angewendet werden können. Beispielsweise können Sie Summen und Mittelwerte für die Zeilen und Spalten der Kalkulationstabelle ermitteln, die Standardabweichung berechnen und statistische Berechnungen mit einer oder zwei Variablen durchführen.

* SPEICHERSCHUTZ — Gespeicherte Programme und Variable bleiben auch nach Ausschalten des Computers erhalten.

* Batteriebetrieb, so daß der Computer jederzeit und überall einsatzbereit ist.

* AUSSCHALT-AUTOMATIK — Nach einer gewissen Zeit ohne Tastenbetätigung schaltet der Computer sich selbsttätig aus, um die Batterien zu schonen.

* Eine erweiterte Version der bei Personal-Computern populären Programmiersprache BASIC bietet formatierte Ausgabe, zweidimensionale Matrizen, Strings variabler Länge und viele andere leistungsstarke Funktionen.

* Als Option sind Drucker und Kassetten-Schnittstellen erhältlich.

Wir gratulieren Ihnen zum Eintritt in eine aufregende und faszinierende neue Welt. Mit diesem Computer haben Sie ein Werkzeug erworben, das Sie bei mathematischen, wissenschaftlichen, technischen, kaufmännischen und persönlichen Anwendungen bald nicht mehr missen wollen. Dieser SHARP-Computer bietet schon heute die Lösungen für morgen.

Bitte beachten Sie, daß für den Betrieb des Computers eine RAM-Karte erforderlich ist. Wenn der Computer eingeschaltet wird, ohne daß eine RAM-Karte eingesetzt ist, funktionieren die Tasten nicht.

KAPITEL 1
VERWENDUNG DIESER BEDIENUNGSANLEITUNG

Diese Bedienungsanleitung soll Sie mit den Möglichkeiten und Merkmalen des Computers vertraut machen und dient gleichzeitig als Nachschlagwerk. Ob Sie ein "Anfänger" oder bereits ein "alter Computer-Hase" sind, sollten Sie auf jeden Fall die Kapitel 2 bis 7 lesen und durcharbeiten.

* Kapitel 2 beschreibt die Hardware Ihres Computers.
* Kapitel 3 erläutert statistische Berechnungen einschließlich der Verwendung von Kalkulationstabellen in der STAT-Betriebsart.
* Kapitel 4 erklärt den Gebrauch des Computers als Taschenrechner.
* Kapitel 5 behandelt Ausdrücke und Begriffe, die für die Programmierung in BASIC von Bedeutung sind, und stellt diese Ausdrücke und Begriffe in Zusammenhang mit der Arbeit mit dem Computer.
* Kapitel 6 dient als Einführung in die BASIC-Programmierung. Eingabe, Korrektur und Ausführung von Programmen werden behandelt.
* Kapitel 7 beschreibt einige Tips und Kniffe, die das Arbeiten mit dem Computer erleichtern.

Kapitel 10 ist ein Nachschlagteil, in dem alle Befehle, Kommandos und Funktionen von BASIC übersichtlich aufgeführt sind.

Erfahrene BASIC-Programmierer können direkt von Kapitel 7 zu Kapitel 10 gehen, um sich mit der BASIC-Implementation dieses Computers vertraut zu machen. Da sich alle BASIC-Dialekte geringfügig unterscheiden, sollte dieses Kapitel zumindest einmal durchgelesen werden, bevor größere Programmieraufgaben in Angriff genommen werden.

Falls Sie bislang noch nicht in BASIC programmiert haben, empfehlen wir den Erwerb eines speziellen Lehrbuchs oder die Teilnahme an einem BASIC-Kurs, bevor Sie diese Kapitel durcharbeiten, da diese Bedienungsanleitung nicht als Lehrbuch für Programmieren konzipiert ist.

Der übrige Teil der Bedienungsanleitung besteht aus:

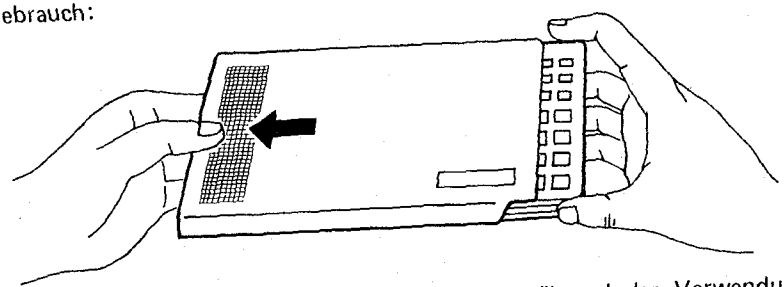
* Kapitel 8 — Grundlegende Informationen über die optionalen Drucker und Kassetten-Schnittstellen.
* Kapitel 9 — Erklärungen zum Gebrauch der RAM-Karte.
* Kapitel 11 — Ein Leitfaden zur Fehlersuche, der bei Schwierigkeiten bei Bedienung und Programmierung helfen soll.
* Kapitel 12 — Pflege und Wartung des Computers.

1.1 TASTATURABDECKUNG

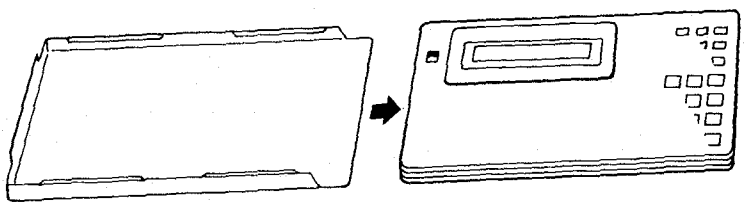
Zum Schutz der Tastatur und der Anzeige ist der Computer mit einer stabilen, doppelseitig aufschiebbaren Tastaturabdeckung ausgestattet.

Die Tastaturabdeckung des Computers wie dargestellt verwenden.

Bei Gebrauch:

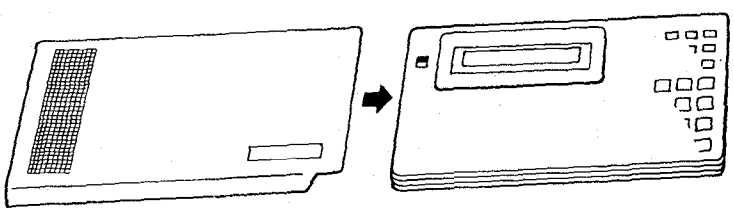


Die Tastaturabdeckung ist so gestaltet, daß sie während der Verwendung des Rechners auf dessen Unterseite geschoben werden kann.



Tastaturabdeckung bei Gebrauch des Rechners

Nach Gebrauch:



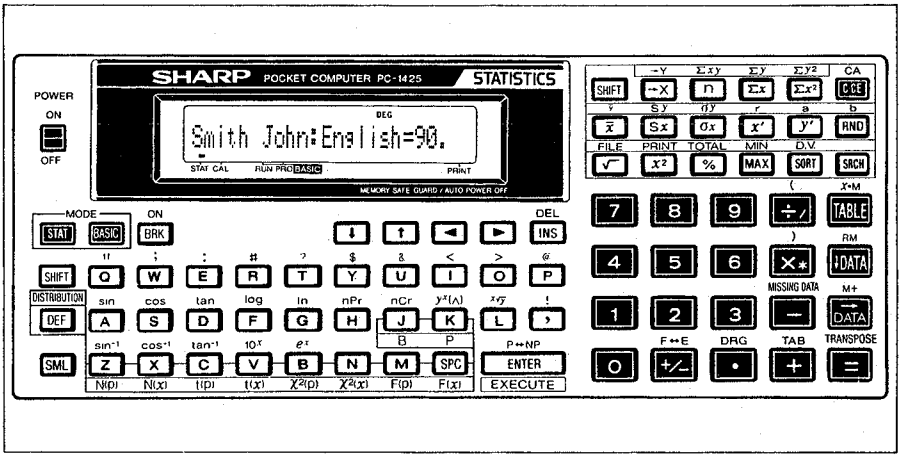
Tastaturabdeckung nach Gebrauch des Rechners

KAPITEL 2
VORSTELLUNG DES COMPUTERS

2.1 ALLGEMEINES

Mit dem **PC-1425** stellt SHARP einen preisgünstigen und dennoch außerordentlich leistungsstarken BASIC-Taschencomputer mit integriertem wissenschaftlichem Rechner vor.

Bitte beachten Sie, daß zum Arbeiten mit dem Computer eine RAM-Karte erforderlich ist. Wenn der Computer eingeschaltet wird, ohne daß eine RAM-Karte eingesetzt ist, funktioniert keine der Tasten.



Frontansicht des **PC-1425**

Zusammen mit dem Thermodrucker mit intergrierterm Kassetten-Interface **CE-126P** oder dem Farbpunkt-Matrixdrucker **CE-140P**/Farbplotter/Drucker **CE-515P** wird der **PC-1425** zu einer kleinen, leistungsfähigen Datenverarbeitungsanlage.

Bei der Gestaltung dieser Bedienungsanleitung wurde versucht, sowohl den Ansprüchen eines Anfängers als auch denen eines geübten Programmierers gerecht zu werden. In manchen Fällen war eine Kompromißlösung aber nicht zu umgehen, und wir verweisen daher bereits an dieser Stelle auf die umfangreiche BASIC-Literatur, die im Fachhandel erhältlich ist.

Wir hoffen, Ihnen mit diesem Handbuch alle möglichen Hilfsmittel in die Hand zu geben, so daß Sie Ihren **PC-1425** für Ihre Belange optimal einsetzen können.

2.2 BEDIENELEMENTE

Das Bedienfeld des **PC-1425** besteht aus 78 Tasten, die zu 4 Blöcken zusammengefaßt sind:

- die Schreibmaschinen-Alphatastatur mit [ENTER] -Taste (2 bzw. 3 Ebenen)
- die Tasten für die Sonderfunktionen
- die numerische Tastatur mit den Grundrechnungsarten (2 Ebenen)
- die Tasten für die statistischen Funktionen (2 Ebenen)

Die Schreibmaschinen-Alphatastatur ist mit zwei Ebenen belegt. Bei einfachem Tastendruck gelten die auf den Tasten stehenden Bezeichnungen. Will man die über oder unter den Tasten stehenden wissenschaftlichen oder Verteilungsfunktionen abrufen, muß vorher die Doppelfunktionstaste [SHIFT] bzw. [DISTRIBUTION] gedrückt werden.

Mit der Taste [DEF] kann den unteren beiden Reihen noch eine dritte Ebene zugewiesen werden. Über die sogenannten 'Definable Keys' können verschiedene Programme sofort gestartet werden; man drückt dazu die Taste [DEF] und eine dem Programm zugewiesene Taste.

In der Reihe über der Alpha-Tastatur befinden sich die Sonderfunktionstasten zur Wahl der Betriebsart, zum Unterbrechen oder zum Auflisten sowie zum Editieren eines Programms.

Rechts neben der Alpha-Tastatur befindet sich die numerische Tastatur. Die zweite Ebene der numerischen Tastatur ist ebenfalls mit der [SHIFT] -Taste erreichbar.

Mit den Tasten oberhalb der numerischen Tastatur lassen sich die meisten statistischen Berechnungen durchführen.

2.3 DIE ANZEIGE

Der Computer verfügt über eine 24-stellige alphanumerische Flüssigkristallanzeige. Die einzelnen Zeichen werden in einer 5 x 7-Punktmatrix dargestellt.

Das Eingaberegister faßt 80 Zeichen. Alle Zeichen, die über die 16 möglichen darstellbaren Zeichen der Anzeige hinausgehen, müssen mit der Cursor-Tasten ([►] und [◄]) 'gescrolled' werden.

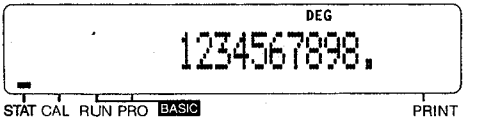
Der Cursor zeigt die Stelle an, an der die nächste Eingabe erfolgen kann. Er steht normalerweise auf der ersten freien Stelle und wird durch einen kurzen waagerechten Strich (_) dargestellt.

Eine Ausnahme besteht, wenn auf der Anzeige links das Bereitschaftssymbol (>) angezeigt wird. Durch das nächste einzugebende Zeichen wird dieses überschrieben. Erst dann sieht man den Cursor auf der ersten freien Stelle nach dem zuletzt eingegebenen Zeichen.

Stellt man den Cursor mit Hilfe der [►] - oder der [◄] -Taste über ein bereits eingegebenes Zeichen, so erkennt man die Position am Blinken des Cursors an dieser Stelle in der vollen Punktmatrix.

Anhaltender Druck auf die Cursor-Tasten läßt den Cursor schnell an die gewünschte Stelle der Zeile bringen. Ein kurzer Druck verschiebt die Anzeige nur um ein Zeichen.

Versucht man mehr als 80 Zeichen einzulesen, werden diese vom Rechner nicht mehr angenommen. Jede zusätzliche Eingabe überschreibt das zuletzt eingegebene Zeichen. Der Cursor verändert hier sein Aussehen; er blinkt auf dem letzten Zeichen in der vollen Punktmatrix.

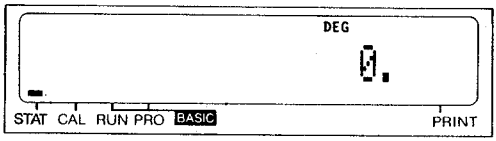


2.4 WAHL DER BETRIEBSART

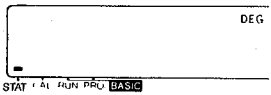
Beim **PC-1425** werden vier Betriebsarten unterschieden:

1. STAT-Mode: Betriebsart für statistische Berechnungen.
2. CAL-Mode: Der Computer wird als wissenschaftlicher Rechner verwendet.
3. RUN-Mode: Gespeicherte Programme können gestartet werden. Zusätzlich kann der Computer im RUN-Mode auch als gewöhnlicher Taschenrechner ohne Programmunterstützung verwendet werden.
4. PRO-Mode: BASIC-Programme können eingegeben, gelistet und verändert werden.

Die Wahl der Betriebsart erfolgt mit den grünen Tasten [STAT] und [BASIC]. Die gewählte Betriebsart wird am Display durch einen kleinen waagerechten Strich (—) angezeigt.



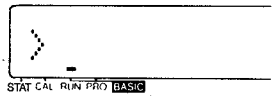
Wird der Computer eingeschaltet, so ist automatisch die Betriebsart "STAT" (statistische Berechnungen) gewählt. Im Display wird das wie dargestellt angezeigt:



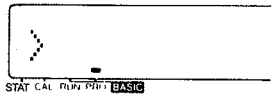
Durch Drücken der Taste [STAT] im STAT-Mode wird in den CAL-Mode geschaltet.

Mit jedem Drücken der Taste [STAT] wird zwischen STAT und CAL umgeschaltet.

Wenn Sie nun die grüne Taste [BASIC] drücken, so wird die Betriebsart von STAT oder CAL in den RUN-Mode umgeschaltet.



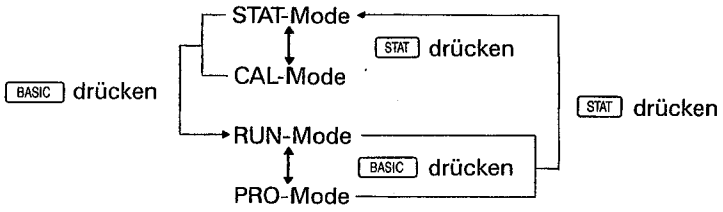
Durch nochmaliges Drücken der Taste [BASIC] wird in den PRO-Mode umgeschaltet.



Die [BASIC] -Taste ist ein Wechselschalter, d.h. durch jedes Drücken wird zwischen RUN und PRO umgeschaltet.

Durch Drücken der [STAT] -Taste kann der Rechner wieder in den STAT-Mode umgeschaltet werden.

Umschaltung zwischen den Betriebsarten



Der STAT-Mode ist die Betriebsart für Arbeiten mit Kalkulationstabellen und statistische Berechnungen.

Die Funktionen einiger Tasten unterscheiden sich in den verschiedenen Betriebsarten. Beachten Sie insbesondere die folgenden Unterschiede:

(1) [X→M] , [RM] und [M+] arbeiten nur in der CAL-Betriebsart.
(2) N(p), N(x), t(p), t(x), χ^2(p), χ^2(x), F(p), F(x), B (unter der Taste "J" angegeben) und P (unter der Taste "K" angegeben) werden in der STAT-Betriebsart berechnet, indem zuerst die Taste [DISTRIBUTION] und danach die entsprechende Taste gedrückt wird.

STAT- und CAL-Mode

Der Computer befindet sich nach dem Einschalten im STAT-Mode. Durch Drücken der Taste [STAT] wird in den CAL-Mode geschaltet und durch erneutes Drücken dieser Taste zurück in den STAT-Mode. Auf diese Weise funktioniert die Taste [STAT] als Wechselschalter für die Betriebsarten STAT und CAL.

Beispiel:

[C·CE]	→	Ø.
[1] [2]	→	12.
[+]	→	12.
[3]	→	3.
[=]	→	15.

RUN- und PRO-Mode

Schalten Sie Ihren Rechner durch Drücken der Taste [BASIC] in den RUN- bzw. in den PRO-Mode.

Beispiele:

[Z] [X] [C]	→	ZXC_
[1] [2] [•] [3]	→	ZXC12.3_
[C·CE]	→	>
[A] [=] [4] [+] [5]	→	A=4+5_ (Cursor ↑)
[C·CE] [SHIFT] [sin⁻¹]	→	ASN _
[SHIFT] [(] [√]	→	ASN (√_

2.5 BETRIEBSHINWEISE

1. Die Flüssigkeitskristallanzeige des Computers ist in Spezialglas eingebettet und kann daher bei Gewalteinwirkung zerbrechen. Behandeln Sie den Computer deshalb mit Sorgfalt. Beim Transport immer die Tastaturabdeckung verwenden.
2. Schützen Sie den Computer vor Staub, Feuchtigkeit und allzu großen Temperaturschwankungen.
3. Zur Reinigung dient ein weiches, trockenes Tuch. Keine Reinigungs- oder Lösungsmittel verwenden.
4. Durch elektrostatische Entladungen über den Computer oder durch Fehlbedienung (der Computer blieb beim Batteriewechsel oder Anschluß der Option **CE-126P** eingeschaltet) kann der Computer "abstürzen". Dadurch werden alle Tastenfunktionen einschließlich der [C·CE] -Taste blockiert.
   Sollte Ihnen dies passieren, müssen Sie den Computer initialisieren.
   Dazu stehen zwei Möglichkeiten zur Auswahl:

FUNKTIONEN DES ALL RESET-SCHALTERS

In seltenen Fällen, wenn der Computer starken elektromagnetischen Störungen oder starken Erschütterungen ausgesetzt war, kann es vorkommen, daß keine der Tasten einschließlich der Taste [BRK] und des Ein/Aus-Schalters funktioniert.
Sollte dies vorkommen, den Ein/Aus-Schalter einschalten und die ALL RESET-Taste in einer der folgenden Weisen verwenden.

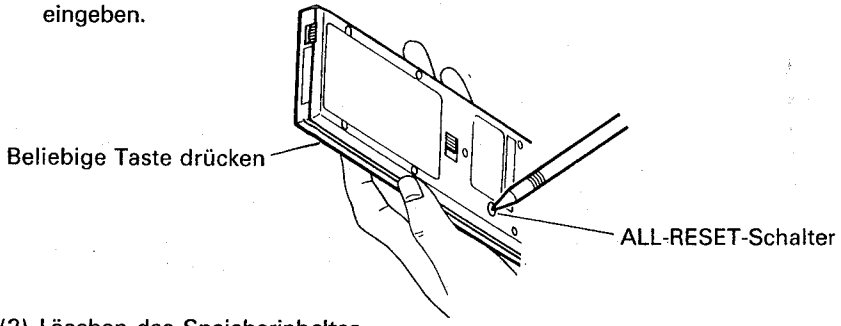
ACHTUNG: Zum Aktivieren der Rückstellfunktion muß der ALL RESET-Schalter (RESET), zumindest 2 — 3 Sekunden lang gedrückt werden. Kürzere Zeitspannen, reichen unter Umständen nicht aus, um den Rechner zu initialisieren. Drücken Sie die All-RESET-Taste erneut, wenn das Symbol * nicht richtig angezeigt wird, die Anzeige verschwindet oder unterschiedlich ist.

Den ALL RESET-Schalter mit einem spitzen Objekt, z. B. einen Kugelschreiber, drücker. Vermeiden Sie Objekte, deren Spitzen abbrechen können (Bleistifte Nadeln) oder breiter als der ALL RESET-Schalter sind.

(1) Erhaltung des Speicherinhaltes

Zum Rückstellen des Computers halten Sie eine beliebige Taste des Computers gedrückt und aktiviereb drücken den ALL RESET-Schalter auf der Rückseite des Rechners mit einem spitzen Objekt. Dadurch wird der Computer zurückgestellt, alle Programme und Variable im Speicher bleiben erhalten.

Sollte das gleiche Problem nach Rückstellung des Computers erneut auftreten, ist es möglich, daß die Ursache im Programm oder in den Daten liegt. In diesem Fall den Computer mit Methode 2 unten löschen und das Programm erneut eingeben.



(2) Löschen des Speicherinhaltes

Nur die ALL RESET-Taste drücken (ohne eine andere Taste gedrückt zu halten). Die folgende Meldung erscheint auf dem Display.

BUSY
MEMORY ALL CLEAR O.K.?

Drücken Sie nun [ENTER], [Y] oder [=].

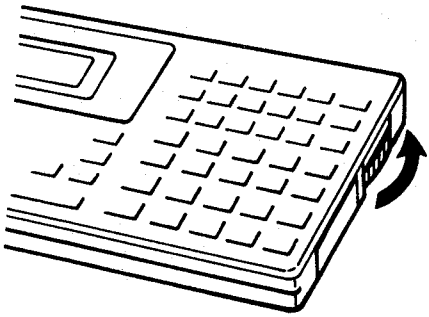
ACHTUNG: Falls innerhalb von ca. 2 Minuten keine dieser Tasten betätigt wird, schaltet sich der Computer automatisch aus.

Durch diesen Initialisierungsvorgang werden alle Inhalte des Speichers (Programm und Daten) und der RAM-Karte gelöscht. Daher sollten Sie den ALL-RESET-Schalter nur dann alleine drücken, wenn eine grundlegende Initialisierung unumgänglich scheint.

Falls das Gerät immer noch nicht richtig funktioniert, entfernen Sie die Batterien für 10 Sekunden. Drücken Sie dann nach erneutem Einsetzen der Batterien den ALL-RESET-Schalter.

KONTRASTREGLER

Ihr Computer besitzt auf der rechten Seite einen Regler, der zum Einstellen des Display-Kontrastes dient. Stellen Sie den Regler auf optimalen Kontrast ein.



Kontrastregler: Durch Drehen des Kontrastreglers in Pfeilrichtung wird der Kontrast der Anzeige verstärkt, in entgegengesetzter Richtung verringert.

2.6 STROMVERSORGUNG

BATTERIEWECHSEL

Der Computer arbeitet ausschließlich mit Lithium-Batterien. Wenn er an die Option **CE-126P** angeschlossen ist, kann der Computer auch über diese versorgt werden, sofern deren momentane Betriebsspannung höher ist. Dies verringert den Stromverbrauch der Lithium-Zellen.

Beim Wechseln der Batterien beachten Sie bitte unbedingt die folgenden Hinweise:

- Wechseln Sie grundsätzlich beide Batterien gleichzeitig.
- Verwenden Sie niemals eine neue gemeinsam mit einer gebrauchten Batterie.
- Benutzen Sie nur Lithium-Zellen (Typ CR-2032, 2 Stück)

Austauschzeitpunkt

Falls die Anzeige auf dem Display trotz Einstellung auf maximalen Kontrast (Kontrastregler ganz nach links drehen) verblassen sollte, ist die Batteriespannung zu niedrig. In diesem Falls sind die Batterien umgehend zu erneuern.

ACHTUNG: Falls Sie das getrennt erhältliche CE-126 Cassetten-Interface oder CE-152 Datenrekorder-Interface einsetzen, können Sie Ihre Programme und Daten im speicher vor dem Auswerchseln der Batterien auf Cassette speichern.

EINSETZEN DER BATTERIEN

Wenn die Anzeige bei maximal eingestelltem Kontrast schwach und schlecht zu erkennen ist, zeigt dies an, daß die Batterien verbraucht sind. In diesem Fall müssen Sie die Batterien umgehend erneuern.

(1) Schalten Sie den Computer ab, indem Sie den Schiebeschalter in die OFF-Position bringen.

(2) Entfernen Sie den rückwärtigen Gehäusedeckel, indem Sie den Verriegelungsknopf (siehe Abbildung 1) in Pfeilrichtung schieben.

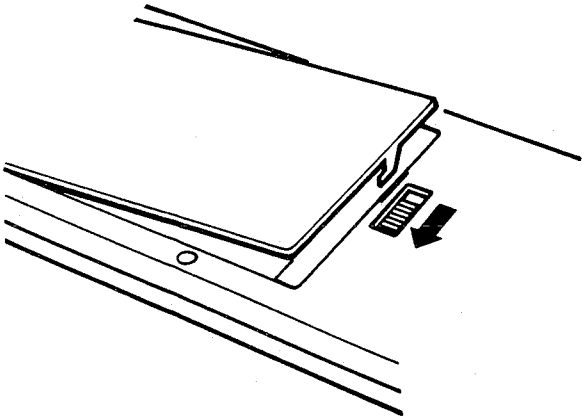


Abb. 1

(3) Wenn Sie eine RAM-Karte benutzen, entfernen Sie diesen wie in Kapitel 9 beschrieben.

(4) Schieben Sie die Batterie-Abdeckung in Pfeilrichtung, sie läßt sich dann abheben.

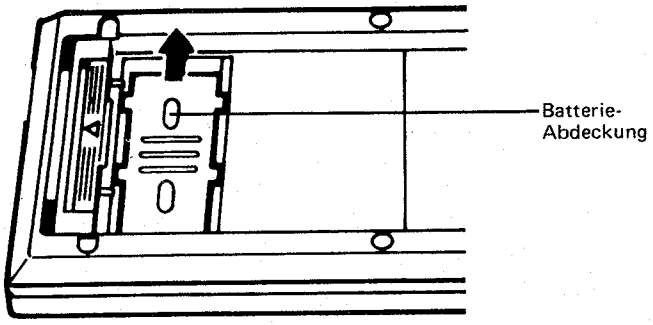


Abb. 2

(5) Erneuern Sie beide Batterien.

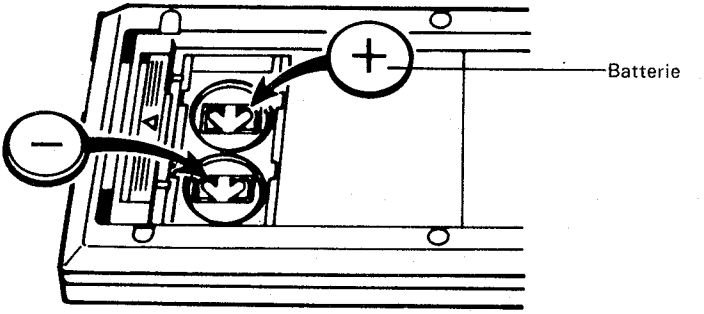
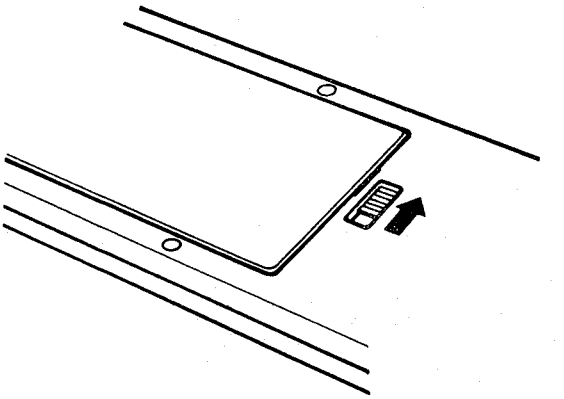


Abb. 3

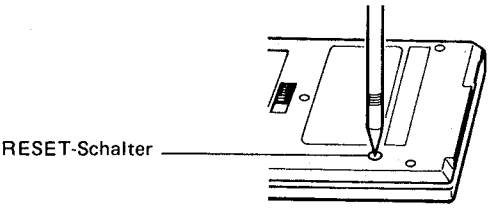
(6) Setzen Sie die Batterie-Abdeckung durch Schieben gegen die Pfeilrichtung wieder ein.

(7) Setzen Sie den rückwärtigen Gehäusedeckel wieder auf und schieben Sie den Verriegelungsknopf wieder in seine ursprüngliche Position.



Beachten Sie: Vergewissern Sie sich, daß der Riegel in der LOCK-Position ist: Der Computer arbeitet sonst nicht. Wenn Sie den Computer einschalten, ohne daß der Riegel in der LOCK-Position ist, schieben Sie den Riegel in die LOCK-Position, schalten den Computer wieder aus und anschließend wieder ein.

(8) Bringen Sie den Schiebeschalter in die Stellung ON und betätigen Sie den RESET-Schalter, um den Computer zu initialisieren.
Drucken Sie die ALL RESET-Taste erneut, wenn das Symbol * nicht richtig angezeigt wird, die Anzeige verschwindet oder unterschiedlich ist.



(9) Das Display muß wie folgt aussehen:

Sollte keine Anzeige auf dem Display ercheinen oder sollten andere Symbole als "*" angezeigt werden, entnehmen Sie die Batterien und setzen sie wieder ein und überprüfen danach die Anzeige erneut.

(10) Den Computer ausschalten und die RAM-Karte einsetzen. (Siehe Kapitel 9.)

ACHTUNG:
Nach Austauschen der Batterien müssen vor Einsetzen der RAM-Karte unbedingt die Schritte (8) und (9) ausgeführt werden. Wird die RAM-Karte eingesetzt, während in Schritt (8) der Computer eingeschaltet ist, wird der Inhalt der RAM-Karte gelöscht.

- **Austauschen der Batterie für RAM-Karte CE-212M**
  Zeitpunkt für Batteriewechsel
  Nach Einsetzen einer Batterie in die RAM-Karte bleibt der Inhalt der RAM-Karte für die folgenden Zeiten erhalten. Die angegebenen Zeiten gelten für eine Lagertemperatur von 20°C.
  - Ca. 5 Jahre, wenn die RAM-Karte im Computer eingesetzt ist.
  - Ca. 15 Monate, wenn die RAM-Karte aus dem Computer entnommen ist.

  Bitte denken Sie daran, vor Ablauf der oben genannten Zeitdauern eine neue Batterie einzusetzen. Schreiben Sie das Datum, an dem Sie die neue Batterie einsetzen, in die Zeile neben dem Batteriefach, so daß Sie wissen, wann Sie die Batterie austauschen müssen.
  Sehr hohe oder niedrige Temperaturen können die Lebensdauer der Batterie verkürzen, so daß der Inhalt der RAM-Karte schon vor Ablauf der oben angegebenen Zeitdauern gelöscht wird. In den folgenden Fällen ist das Austauschen der Batterien erforderlich.

1. Programme im Speicher können nicht ausgeführt werden.
2. Auf dem Display erscheinen bedeutungslose Daten und Anzeigen.
3. Es treten häufig Fehler ohne erkennbare Ursache auf.

Auswechseln der Batterie

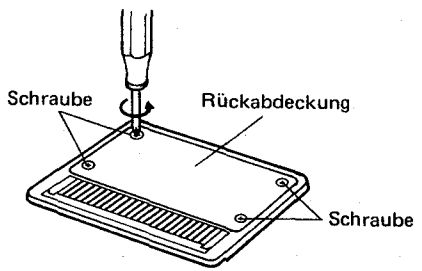
In diesem Abschnitt wird das Auswechseln der Batterie der getrennt erhältlichen CE-212M RAM-Karte erläutert.

Achtung:

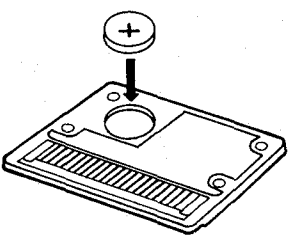
Vor Auswechseln der Batterie sollte der Inhalt der RAM-Karte auf Cassette aufgenommen werden. ALLE DATEN UND PROGRAMME WERDEN GELÖSCHT, WENN DIE BATTERIE AUS DER RAM-KARTE ENTNOMMEN WIRD.

- Die Batterien in den RAM-Karten von Standardgröße können ausgewechselt werden, während die Karte im Computer eingesetzt ist. Die Inhalte der RAM-Karten werden durch den vom Computer gelieferten Strom aufrecht erhalten. *(Siehe Kapitel 9 über RAM-Karten).*

(1) Die vier Schrauben mit einem Kreuzschlitzschraubenzieher entfernen und die Rückabdeckung entfernen.

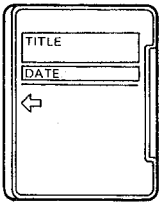


(2) Die alte Batterie entnehmen. Die neue Batterie mit einem weichen, trockenen Tuch abwischen und einsetzen.



(3) Die Rückabdeckung wieder anbringen und die Schrauben anziehen.

(4) Das Datum mit unlöschbarer Tinte in die Zeile DATE schreiben.



- Die Namen der in der RAM-Karte gespeicherten Programme in die Zeile TITLE schreiben.
- Die RAM-Karte wieder in den Computer einsetzen und die vorher auf Cassette gespeicherten Programme einlesen.

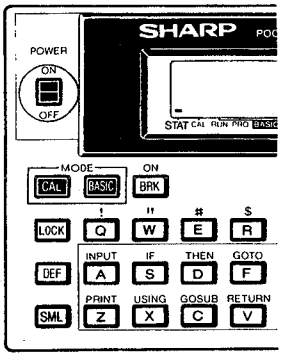
Achtung: Läßt man eine leere Batterie im Computer, kann er durch Säureaustritt aus der Batterie geschädigt werden. Leere Batterien umgehend erneuern!

Vorsicht: Bewahren Sie die Batterien außerhalb der Reichweite von Kindern auf!

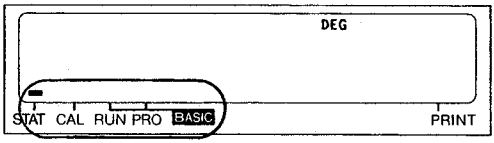
KAPITEL 3
STATISTISCHE BERECHNUNGEN

3.1 EINSCHALTEN DES COMPUTERS

Das Einschalten des Computers erfolgt über den Schiebeschalter links neben dem Display.
Der Computer befindet sich nach dem Einschalten in der STAT-Betriebsart. Dann leuchtet ein Strich (−) über dem Symbol "STAT" unten links auf dem Display.



Nach dem Einschalten erschalten erscheinen auf der Anzeige folgende Symbole:



Zeigt Betriebsart (STAT/CAL/RUN/PRO)

Wenn der Strich (▬) sich über dem Symbol RUN oder PRO befindet, die Taste [STAT] zweimal drücken. Ist der Strich (▬) über dem Symbol CAL, die Taste [STAT] einmal drücken.

3.2 AUSSCHALTEN DES COMPUTERS

Automatisch

Der Computer schaltet sich nach ca. 11 Minuten nach der letzten Tastenbedienung automatisch ab. Dabei bleiben alle Programm- und Dateninformationen gespeichert.

Hat sich der Rechner automatisch ausgeschaltet, muß er über die Taste ON/BRK wieder aktiviert werden. Nach dem Einschalten befindet sich der Rechner in der Betriebsart "STAT".

Manuell

Soll der Computer sofort nach Gebrauch ausgeschaltet werden, so muß der Schiebeschalter links neben dem Display in Stellung OFF gebracht werden. Die Programm- und Dateninformationen bleiben dabei gespeichert. Der Computer läßt sich auch während einer Programmausführung abschalten.

3.3 BETRIEBSART FÜR STATISTISCHE BERECHNUNGEN

Es gibt 2 Betriebsarten:

- Kalkulationstabellen-Berechnungen
- Statistische Berechnungen mit einer oder zwei Variablen
- Verteilungsfunktion

- Bei Kalkulationstabellen-Berechnungen werden Daten in die Kalkulationstabelle eingegeben und gespeichert. Für jede Zeile und Spalte können Summen- und Mittelwert-Berechnungen durchgeführt werden. Außerdem kann die Abweichung und der Prozentsatz (%) für jede Spalte berechnet werden.

- Bei den Berechnungen mit einer oder zwei Variablen werden die Daten, die für die Kalkulationstabellen-Berechnungen eingegeben wurden, für die verschiedenen statistischen Berechnungen verwendet.

- Mit der Verteilungsfunktion kann das obere Wahrscheinlichkeitsintegral für die Normalverteilung, t-Verteilung und andere Verteilungen ermittelt werden.

Verwendung der statistischen Berechnungsfunktionen

Zur Verwendung des Computers für statistische Berechnungen muß zuerst die STAT-Betriebsart eingestellt werden. In dieser Betriebsart wird das Symbol ▬ unten links auf dem Display, wo "STAT" geschrieben ist, angezeigt. Wenn das Symbol ▬ an einer anderen Position angezeigt wird, die Taste [STAT] drücken.

3.4 KALKULATIONSTABELLEN-BETRIEBSART

In der STAT-Betriebsart werden Daten in die Kalkulationstabelle eingegeben und gespeichert. Dann kann die Summe und der Mittelwert für jede Spalte und Zeile berechnet werden. Außerdem kann die Abweichung und der Prozentsatz (%) für jede Spalte ermittelt werden. Die beiden letztgenannten Berechnungen können zwar nur für die Spalten der Kalkulationstabelle ausgeführt werden, aber mit der Taste [TRANSPOSE] können die Spalten und Zeilen vertauscht werden. Das bedeutet, daß die statistischen Berechnungen auch auf die Zeilen angewendet werden können, indem zuerst die Zeilen und Spalten vertauscht und dann die Berechnungen für die Spalten durchgeführt werden.
Geben Sie zur Übung Daten für die statistischen Berechnungen in die Kalkulationstabelle ein. Zuerst geben Sie den Titel der Kalkulationstabelle sowie die Spalten- und Zeilen-Namen und danach die numerischen Daten ein, siehe Tabelle 3-1.

Tabelle 3-1
Titel
SCORES

	MATH	SCIENCE	ENGLISH	HISTORY ← Spalten-Namen
ADAMS	80.	90.	60.	50.
BOND	20.	30.	25.	69.
JOHNSON	39.	41.	63.	38.
THOMPSON	51.	22.	60.	70.
YOUNG	90.	80.	100.	70.

↑ Zeilen-Namen

Nach Eingabe der Daten wollen wir die Abweichung ermitteln und statistische Berechnungen mit einer Variablen durchführen.

1 Eingabe des Titels

Drücken Sie die Taste [TABLE], wodurch in die Kalkulationstabellen-Betriebsart geschaltet wird. Die folgende Anzeige erscheint:

```
TABLE TITLE 1=?
```

Geben Sie den Titel nach dem Fragezeichen (?) ein. Ohne Eingabe eines Titels kann nicht weitergemacht werden. Der Titel kann aus bis zu 16 alphanumerischen Zeichen bestehen. Sie können bis zu 10 Titel von Nummer 1 bis Nummer 10 eingeben. Drükken Sie die Taste [TABLE] erneut. Die folgende Anzeige erscheint:

```
TABLE TITLE 2=?
```

Drücken Sie jetzt die Tasten [SHIFT] [TABLE]. Die folgende Anzeige erscheint, wobei die Nummer des Titels um eins verringert wird:

```
TABLE TITLE 1=?
```

Werden die Tasten [SHIFT] [TABLE] bei Titel Nummer 1 gedrückt, ändert sich die Titel-Nummer zu 10.

```
TABLE TITLE 10=?
```

Die Titel-Nummern bilden eine Schleife mit 1 vor 2 und 10 vor 1.
Geben Sie bei der Anzeige

```
TABLE TITLE 1=?
```

den Titel nach dem Fragezeichen (?) ein. Wenn Sie SCORES eingeben, wird

```
TABLE TITLE 1=SCORES _
```

angezeigt. Wenn Sie jetzt die Taste [DATA] drücken, wird

```
COLUMN ITEM 1=?
```

angezeigt, so daß Sie den Spalten-Namen eingeben können.

- Zur Korrektur einer Eingabe wird entweder der Cursor mit der Taste [◄] um ein Zeichen zurückbewegt und dann über das Zeichen geschrieben oder die Eingabe mit der Taste [C·CE] gelöscht und erneut durchgeführt. Wenn eine Fehlermeldung angezeigt wird, kann sie mit der Taste [C·CE] gelöscht werden, dann wird das ursprüngliche Feld wieder angezeigt.

2. Eingabe der Spalten-Namen

Wie beim Titel im vorigen Abschnitt wird

```
COLUMN ITEM 1=?
```

angezeigt, und Sie können den Spalten-Namen eingeben. Nach Eingabe von MATH wird

```
COLUMN ITEM 1=MATH _
```

angezeigt. Durch Drücken der Taste [DATA] wird

```
COLUMN ITEM 2=?
```

angezeigt. Geben Sie die in Tabelle 3-1 enthaltenen Spalten-Namen durch Betätigung von SCIENCE [DATA] ENGLISH [DATA] HISTORY [DATA] ein.

```
COLUMN ITEM 5=?
```

Beendigung der Eingabe der Spalten-Namen

Drücken Sie nach Eingabe aller Spalten-Namen erneut die Taste [DATA]. Dann erscheint eine neue Anzeige, und Sie können die Zeilen-Namen eingeben.

```
ROW ITEM 1=?
```

Die Spalten-Namen müssen eingegeben werden.

Hinweis: Der Titel der Kalkulationstabelle sowie die Zeilen- und Spalten-Namen können mit einem beliebigen Zeichen außer Leerstelle beginnen. Die Länge kann in allen Fällen bis zu 16 alphanumerischen Zeichen betragen.

3. Eingabe der Zeilen-Namen und Daten

Nach Eingabe der Spalten-Namen muß die Anzeige

ROW ITEM 1=?

abgerufen werden, um die Zeilen-Namen einzugeben. Durch Eingabe von ADAMS wird

ROW ITEM 1=ADAMS _

angezeigt. Durch Drücken der Taste [DATA] wird

ADAMS:MATH=?

angezeigt.
Wenn Sie keine Zeilen-Namen eingeben, werden die Nummern

001, 002, ...

automatisch anstelle der Zeilen-Namen zugewiesen.
Nach dem obigen Schritt wird

ADAMS:MATH=?

angezeigt. Durch die Eingabe von 80 wird

ADAMS:MATH=80 _

angezeigt. Wenn Sie jetzt die Taste [DATA] drücken, wird

ADAMS:SCIENCE=?

angezeigt. Dies bedeutet, daß die Mathematik-Punktzahl von Adams gespeichert ist. Durch die Eingabe von 90 [DATA] 60 [DATA] 50 [DATA] werden die Punktzahlen von Adams für die 4 Fächer gespeichert, und

ROW ITEM 2=?

wird angezeigt. Die Punktzahlen von Bond werden durch die Eingabe BOND [DATA] 20 [DATA] 30 [DATA] 25 [DATA] 69 [DATA] gespeichert. Geben Sie auf die gleiche Weise die Punktzahlen für JOHNSON, THOMPSON und YOUNG aus Tabelle 3-1 ein. Zuletzt wird

ROW ITEM 6=?

angezeigt. Damit ist die Eingabe der Daten aus Tabelle 3-1 beendet. Die Position, die durch die Spalten- und Zeilen-Nummer bestimmt ist, wird als Feld bezeichnet. Mit den Tasten [►], [◄], [↑] und [↓] können Sie die Felder durchgehen und überprüfen, daß die Anzeigen mit dem Inhalt von Tabelle 3-1 übereinstimmen.

Hinweis: Eine zusätzliche Zeile ist enthalten, um bei Bedarf eine weitere Zeile hinzufügen zu können.

- Es ist auch möglich, zuerst alle Zeilen-Namen anstatt gleichzeitig mit den Daten einzugeben. Dafür dient die Taste [↓DATA], zum Beispiel: [↓DATA] JOHNSON [↓DATA] THOMPSON [↓DATA] YOUNG [↓DATA].

Unmittelbar nach Eingabe der Zeilen-Namen können die Summen und Mittelwerte berechnet sowie die Felder durchgangen werden.

Korrektur von Daten

Wenn der Cursor nicht angezeigt wird, können die angezeigten Daten (einschließlich Titel sowie Spalten- und Zeilen-Namen) korrigiert werden, indem einfach die richtigen Daten eingegeben werden. Um nur einen Teil der Daten zu korrigieren, die Taste [C·CE] drücken, wenn die zu korrigierenden Daten angezeigt werden. Nachdem der Cursor angezeigt wird, den Cursor mit den Tasten [◄] und [►] zu der Stelle bewegen, die geändert werden soll, und die richtigen Zeichen eingeben bzw. die erforderlichen Einfügungen oder Löschungen durchführen. Nach jeder Korrektur muß die Taste [DATA] gedrückt werden, um die Korrektur in den Speicher zu schreiben. Die Speicherung einer Korrektur erfolgt nur, wenn die Taste [DATA] oder [↓DATA] gedrückt wird.

- Die Daten können jeden Wert im Bereich $0 \leqq |\text{Daten}| < 1 \times 10^{99}$ annehmen. Wenn dieser Bereich verlassen wird, tritt ein Fehler auf. Wird dieser Bereich bei der Berechnung einer Summe verlassen, wird im Summen- und Mittelwert-Feld "****" angezeigt.
  Die Daten für statistische Berechnungen mit einer oder zwei Variablen müssen jedoch im Bereich $0 \leqq |\text{Daten}| < 1 \times 10^{50}$ liegen. Sonst tritt ein Fehler auf.

- Das Ergebnis einer Berechnung oder Funktion kann als Datenwert eingegeben werden.

Beispiel: 100/10 [DATA] SIN30 [DATA]

- Zur Eingabe von fehlenden Daten die Tasten [MISSING DATA] [DATA] drücken, während der Cursor nicht angezeigt wird. Wenn ein Datenwert noch nicht vorliegt, die Tasten [SHIFT] [?] [DATA] drücken. Fehlende Datenwerte werden als "- - -" und nicht vorliegende Datenwerte als "?" angezeigt. In den nachfolgenden Berechnungen werden fehlende und nicht vorliegende Daten ignoriert.

Verlassen der Kalkulationstabellen-Betriebsart
Durch Drücken der Tasten [STAT] oder [SHIFT] [CA] kann die Dateneingabe in der Kalkulationstabellen-Betriebsart verlassen werden.

Bytes-Anzahl der Kalkulationstabellen-Daten
Die Kapazität (in Bytes) einer Kalkulationstabelle ergibt sich durch die folgende Formel:
[32[1)] + 32 × (Anzahl der Spalten + Anzahl der Zeilen) + 8 × (Anzahl der Spalten + 2) × (Anzahl der Zeilen + 2) + 21[2)]] Bytes

32 Bytes werden für den Titel[1)] der Kalkulationstabelle und 21 Bytes für die Feld-Deklaration[2)] benötigt.

Beispielsweise ermöglicht eine 8 KB RAM-Karte Berechnungen mit drei Kalkulationstabellen mit je 10 Spalten und 10 Zeilen, eine 16 KB RAM-Karte mit acht Kalkulationstabellen mit je 10 Spalten und 10 Zeilen und eine 32 KB RAM-Karte mit zehn Kalkulationstabellen mit je 10 Spalten und 20 Zeilen. Dabei wird jedoch davon ausgegangen, daß der Computer ausschließlich zur Speicherung der Kalkulationstabellen-Daten eingesetzt wird. Die maximale Anzahl der Zeilen und der Spalten ist jeweils 250.

4. Beschreibung der wichtigsten Tasten

TABLE:
Bei Drücken ohne die Taste [SHIFT]:
In der STAT-Betriebsart wird in die Kalkulationstabellen-Betriebsart geschaltet.
In der Kalkulationstabellen-Betriebsart wird der nächste Kalkulationstabellen-Titel angezeigt.

[TABLE]

```
TABLE TITLE 1=SCORES
```

```
TABLE TITLE 2=?
```

Nach Titel 1 folgt Titel 10.

	TABLE TITLE 10=?
TABLE	TABLE TITLE 1=SCORES

Wenn jedoch andere Daten als der Titel angezeigt werden, kehrt das Display zur Anzeige des Titels zurück.
Bei Drücken mit der Taste SHIFT :
In der STAT-Betriebsart wird in die Kalkulationstabellen-Betriebsart geschaltet.
In der Kalkulationstabellen-Betriebsart wird der vorige Titel angezeigt.

	TABLE TITLE 1=SCORES
TABLE	TABLE TITLE 2=?
SHIFT TABLE	TABLE TITLE 1=SCORES

Nach Titel 10 folgt Titel 1.

	TABLE TITLE 1=SCORES
SHIFT TABLE	TABLE TITLE 10=?

C·CE:
Durch Drücken, während der Cursor nicht angezeigt wird, erscheint der blinkende Cursor an der linken Seite der Eingabedaten, die dann korrigiert werden können.
Durch Drücken, während der Cursor angezeigt wird, verschwindet der Cursor, und die in dem Feld bereits gespeicherten Daten werden angezeigt.

▶:
Durch Drücken, während der Cursor angezeigt wird und rechts vom Cursor ein Zeichen vorhanden ist, bewegt sich der Cursor um ein Zeichen nach rechts.
Durch Drücken, während der Cursor nicht angezeigt wird, wird das Feld rechts vom gegenwärtig angezeigten Feld angezeigt.

	TABLE TITLE 1=SCORES
▶	COLUMN ITEM 1=MATH

BOND:SCIENCE=30.

[►]

BOND:ENGLISH=25.

- Wenn der Cursor die rechte Seite erreicht, erfolgt keine weitere Aktion.
- Durch Drücken zusammen mit der Taste [SHIFT] wird das ganz rechte Feld angezeigt.
- Wenn der Inhalt des Displays 24 Stellen überschreitet, wird rechts das Symbol → angezeigt. Durch Drücken der Taste [►] wird der Teil an der rechten Seite angezeigt.

◄:

Durch Drücken, während der Cursor angezeigt wird und sich links ein Zeichen befindet, bewegt sich der Cursor um ein Zeichen nach links.
Durch Drücken, während der Cursor nicht angezeigt wird, wird das Feld links vom gegenwärtig angezeigten Feld angezeigt.

BOND:ENGLISH=25.

[◄]

BOND:SCIENCE=30.

- Wenn der Cursor sich im Titel der Kalkulationstabelle befindet, erfolgt keine weitere Aktion.
- Wenn der Cursor die rechte Seite erreicht, erfolgt keine weitere Aktion.
- Durch Drücken zusammen mit der Taste [SHIFT] wird das ganz linke Feld angezeigt.
- Wenn der Inhalt des Displays 24 Stellen überschreitet und die Taste [►] gedrückt wird, um den Teil an der rechten Seite anzuzeigen, wird links das Symbol ← angezeigt. Durch Drücken der Taste [◄] erfolgt Anzeige von der ersten Stelle an.

↓:

Zeigt das Feld unter dem gegenwärtig angezeigten Feld an.

BOND:SCIENCE=30.

[↓]

JOHNSON:SCIENCE=41.

- Durch Drücken zusammen mit der Taste [SHIFT] wird die unterste Zeile angezeigt.
- Bei Erreichen der untersten Zeile erfolgt keine weitere Aktion.

↑:
Zeigt das Feld über dem gegenwärtig angezeigten Feld an.

	JOHNSON:SCIENCE=41.
[↑]	BOND:SCIENCE=30.

- Durch Drücken zusammen mit der Taste [SHIFT] wird die oberste Zeile angezeigt.
- Bei Erreichen der obersten Zeile erfolgt keine weitere Aktion.

$\vec{\text{DATA}}$:
Gibt Daten in das gegenwärtig angezeigte Feld ein und zeigt den Inhalt des nächsten Feldes an der rechten Seite an.

	JOHNSON:SCIENCE=41.
52 [$\vec{\text{DATA}}$]	JOHNSON:ENGLISH=63.
[◄]	JOHNSON:SCIENCE=52.

Wenn jedoch eine Eingabe an der ganz rechten Seite erfolgt, wird der Zeilen-Name der nächsten Zeile angezeigt.

↓DATA:
Gibt Daten in das gegenwärtig angezeigte Feld ein und zeigt den Inhalt des Feldes eine Zeile tiefer in der gleichen Spalte an.

	JOHNSON:SCIENCE=41.
52 [↓DATA]	THOMPSON:ENGLISH=22.
[↑]	JOHNSON:SCIENCE=52.

3.5 EDITIERFUNKTIONEN

1. Anzeige von Daten

In der STAT-Betriebsart die Taste [TABLE] drücken, um die Kalkulationstabellen-Betriebsart einzustellen.

```
TABLE TITLE 1=SCORES
```

Der vorher gespeicherte Titel wird angezeigt, siehe oben. Wenn kein Titel angezeigt wird, d.h. wenn

```
TABLE TITLE 1=?
```

angezeigt wird, suchen Sie durch Drücken der Taste [TABLE] nach einer Kalkulationstabelle mit gespeichertem Titel oder geben Sie die auf Seite 25 aufgeführten Daten erneut ein.
Wenn der Titel angezeigt wird, kann der Spalten-Name mit der Taste [►] oder [DATA] und der Zeilen-Name mit der Taste [↓] oder [ENTER] angezeigt werden. Nach der Beschreibung auf Seite 30 können die in den Feldern gespeicherten Daten angezeigt werden.
Zur Korrektur von Daten Siehe 28.

2. Löschen

Wenn der Kalkulationstabellen-Titel angezeigt wird:
Drücken Sie die Tasten [SHIFT] [DEL]. Die folgende Anzeige erscheint:

```
DELETE 1:TABLE 2:DATA
```

Durch Eingabe von 1 wird die gesamte Kalkulationstabelle und durch Eingabe von 2 werden nur die Datenwerte in der Kalkulationstabelle gelöscht. Wenn Sie eine andere Taste drücken, wird wieder der Titel der Kalkulationstabelle angezeigt.
Wenn ein Feld einer Datenzeile angezeigt wird:
Durch Drücken der Tasten [SHIFT] [DEL] wird die Zeile gelöscht, die das gegenwärtig angezeigte Feld enthält.

3. Einfügen

Wenn die Taste [INS] gedrückt wird, während ein Feld einer Datenzeile angezeigt wird und weniger als 250 Datenzeilen vorhanden sind, wird unmittelbar über der Zeile, die das gegenwärtig angezeigte Feld enthält, eine Zeile eingefügt.

Hinweis: Die Taste [INS] wird in den folgenden Fällen ignoriert:

1. Wenn bereits 250 Datenzeilen vorhanden sind.
2. Wenn der Kalkulationstabellen-Titel oder ein Spalten-Name angezeigt wird.

4. Transposition

Zum Löschen oder Einfügen einer Spalte müssen die Zeilen und Spalten zuerst mit der Taste [TRANSPOSE] vertauscht werden, dann kann die Löschung oder Einfügung durchgeführt werden. Nach der Löschung bzw. Einfügung die Spalten und Zeilen wieder mit der Taste [TRANSPOSE] vertauschen.

3.6 GRUNDLEGENDE KALKULATIONSTABELLEN-FUNKTIONEN

1. Verwendung der Berechnungsfunktionen

Elemente, die länger als 24 Stellen sind, werden in 2 Teilen angezeigt. Wenn an der rechten Seite das Symbol → angezeigt wird, kann der Teil an der rechten Seite mit der Taste [►] angezeigt werden. Wird an der linken Seite das Symbol ← angezeigt, erfolgt durch Betätigung der Taste [◄] die Anzeige von der ersten Stelle an.

Prozentsatz

Durch Drücken der Taste [%] wird der Anteil des gegenwärtig angezeigten Datenwerts zur Spaltensumme als Prozentsatz angezeigt.

	THOMPSON:ENGLISH=60.
[%]	←P.C.%=19.48051948
[◄]	THOMPSON:ENGLISH:P.C.%=→
[►]	←P.C.%=19.48051948

Wenn ein Drucker angeschlossen ist, können dann die Prozentsätze der gesamten Spalte durch Drücken der Tasten [SHIFT] [PRINT] ausgedruckt werden.
Durch Drücken von [C·CE], [↓], [↑], [DATA] oder [↓DATA] wird die Prozentsatz-Anzeige verlassen.

Summe/Mittelwert

Durch Drücken der Taste [TOTAL] werden die Summen und Mittelwerte für alle Zeilen und Spalten gleichzeitig berechnet. Felder für die Summen und Mittelwerte werden in der Kalkulationstabelle bereitgestellt und mit den Werten gefüllt.

Tabelle 3-2
SCORES

	MATH	SCIENCE	ENGLISH	HISTORY	ROW TOTAL	ROW MEAN
ADAMS	80.	90.	60.	50.	280.	70.
BOND	20.	30.	25.	69.	144.	36.
JOHNSON	39.	41.	63.	38.	181.	45.25
THOMPSON	51.	22.	60.	70.	203.	50.75
YOUNG	90.	80.	100.	70.	340.	85.
COLUMN TOTAL	280.	263.	308.	297.	1148.	— — — —
COLUMN MEAN	56.	52.6	61.6	59.4	— — — —	— — — —

COLUMN TOTAL: Spalten Summe ROW TOTAL: Zeilen Summe
COLUMN MEAN: Spalten Mittelwert ROW MEAN: Zeilen Mittelwert

Die obige Tabelle enthält die Summen und Mittelwerte aller Zeilen und Spalten, die mit den Kalkulationstabellen-Berechnungsfunktionen ermittelt wurden.

[SHIFT] [TOTAL] **PLEASE PUSH ENTER**

[EXECUTE] ***EXECUTING***

Nach Durchführung der Berechnungen erscheint wieder die Anzeige wie vor dem Drücken der Taste [TOTAL]. Durch Abrufen des Feldes unter der untersten Zeile mit der Taste [↓] wird die Summe dieser Spalte angezeigt. Mit der Taste [►] können die Summen der Spalten nach rechts und mit der Taste [◄] die Summen der Spalten nach links erreicht werden. Durch Betätigung der Taste [↓] wird der Mittelwert der jeweiligen Spalte angezeigt. Wie bei den Summen können die Mittelwerte der Spalten nach rechts und links mit den Tasten [►] bzw. [◄] angezeigt werden.

[↓]	YOUNG:MATH=90.
[↓]	006:MATH=?
[↓]	MATH SUM=280.
[►]	SCIENCE SUM=263.
[↓]	SCIENCE MEAN=52.6

Wenn nach Berechnung von Summen/Mittelwerten ein neuer Datenwert in ein Feld eingegeben wird, werden die Werte für die Summen und Mittelwerte gelöscht und können dann durch Drücken der Taste [TOTAL] neuberechnet werden.

Abweichung

Durch Drücken der Taste [D.V.] wird die Abweichung des Wertes des gegenwärtig angezeigten Feldes für die Spalte, in der sich das Feld befindet, berechnet.

	ADAMS:SCIENCE=90.
[SHIFT] [D.V.]	D.V.=63.69091077
[◄]	ADAMS:SCIENCE:D.V.=63.6→

Wenn ein Drucker angeschlossen ist, können die Abweichungen der gesamten Spalte druch Drücken der Tasten [SHIFT] [PRINT] ausgedruckt werden. Ist nur ein Datenwert vorhanden, erfolgt keine Berechunug, da die Abweichung dann nicht berechnet werden kann.

Zum Verlassen der Abweichungs-Anzeige [C·CE], [↓], [↑], [DATA] oder [↓DATA] drücken.

Maximum

Durch Drücken der Taste [MAX] wird der Maximalwert der Spalte, in der sich das gegenwärtig angezeigte Feld befindet, angezeigt. Enthalten mehrere Felder der gleichen Spalte diesen Maximalwert, wird der Maximalwert nur für ein Feld angezeigt.

	ADAMS:ENGLISH=60.
[MAX]	YOUNG:ENGLISH=100.

Minimum

Minimum Durch Drücken der Taste [MIN] wird der Minimalwert der Spalte, in der sich das gegenwärtig angezeigte Feld befindet, angezeigt. Enthalten mehrere Felder der gleichen Spalte diesen Minimalwert, wird der Minimalwert nur für ein Feld angezeigt.

	ADAMS:ENGLISH=60.
[SHIFT] [MIN]	BOND:ENGLISH=25.

Zufallszahlen

Durch Drücken der Taste [RND] wird eine Pseudo-Zufallszahl von 0.000 bis 0.999 (z.B. 0.178) erzeugt und rechts auf dem Display angezeigt. In der Statistik werden häufig Zufallszahlen für partielle Stichproben verwendet.

	BOND:ENGLISH=25.
[RND]	BOND:ENGLISH=25. [0.929]

2. Vertauschen von Spalten und Zeilen

Die auf Seite 35 beschriebenen Funktionen können mit Ausnahme der Summenund Mittelwert-Berechnung nur für Spalten ausgeführt werden. Zur Durchführung dieser Funktionen für Zeilen müssen zuerst die Spalten und Zeilen mit der Taste [TRANSPOSE] vertauscht werden.

[SHIFT] [TRANSPOSE]	PLEASE PUSH ENTER
[EXECUTE]	***EXECUTING***

3. Exponential- und Fließkomma-Darstellung

TAB:
Nach Drücken der Taste [TAB] kann die Anzahl der Dezimalstellen speizifiziert werden. Zwischen 0 und 9 Dezimalstellen können spezifiziert werden.

	JOHNSON MEAN=45.25
[SHIFT] [TAB] [3]	JOHNSON MEAN=45.250
[↓]	THOMPSON MEAN=50.750

- Die Taste [TAB] dient zur Spezifizierung der Anzahl Dezimalstellen der Berechnungsergebnisse und beeinflußt die eingegebenen Daten nicht.
- Zum Aufheben der Anzahl Dezimalstellen die Tasten [SHIFT] [TAB] [•] drücken.

F↔E:
Durch Drücken der Taste [F↔E] schaltet die Anzeige auf von Exponential- auf Fließkomma-Darstellung bzw. von Fließkomma- auf Exponential-Darstellung um.

	THOMPSON MEAN=50.75
[SHIFT] [F↔E]	THOMPSON MEAN=5.075E 01

4. Sortieren

Die Werte in der Spalte, zu dem das gegenwärtig angezeigte Feld gehört, können sortiert werden. Sortieren ist entweder in aufsteigender oder absteigender Reihenfolge möglich.

[SORT]	1:MAX→MIN 2:MIN→MAX
2	***EXECUTING***

spezifiziert aufsteigende Reihenfolge

1 spezifiziert Sortieren in absteigender und 2 in aufsteigender Reihenfolge. Zum Schutz der Daten funktionieren der Ein/Aus-Schalter und die Taste [BRK] nicht, bis der Sortiervorgang beendet ist. Wird eine andere Taste als 1 oder 2 betätigt, erfolgt kein Sortieren, dann werden die Daten des ursprünglichen Feldes angezeigt.
Nach dem Sortieren werden die Daten des Feldes ganz oben in der Spalte angezeigt.

Tabelle 3-3
SCORES

	MATH	SCIENCE	ENGLISH	HISTORY	ROW TOTAL	ROW MEAN
BOND	20.	30.	25.	69.	144.	36.
JOHNSON	39.	41.	63.	38.	181.	45.25
THOMPSON	51.	22.	60.	70.	203.	70.75
ADAMS	80.	90.	60.	50.	280.	50.75
YOUNG	90.	80.	100.	70.	340.	85.
COLUMN TOTAL	280.	263.	308.	297.	1148.	— — — —
COLUMN MEAN	56.	52.6	61.6	59.4	— — — —	— — — —

Tabelle 3-3 zeigt das Ergebnis des Sortierens nach den Mathematik-Punktzahlen der Tabelle 3-2 in aufsteigender Reihenfolge, das bei Anzeige des Feldes THOMPSON:MATH durchgeführt wurde.

5. Suchen

Wenn unter den eingegebenen Daten ein bestimmter Datenwert gesucht wird, bewegt sich der Cursor zum ersten gleichen Datenwert der Spalte, von dem gegenwärtig ein Feld angezeigt wird. Zum Fortsetzen der Suche die Tasten [SHIFT] [SRCH] drücken. Sind keine übereinstimmenden Daten vorhanden, wird der Spalten-Name dieser Spalte angezeigt.

	BOND:HISTORY=69.
[SRCH]	SEARCH:?
70 [EXECUTE]	THOMPSON:HISTORY=70.
[SHIFT] [SRCH]	YOUNG:HISTORY=70.
[SHIFT] [SRCH]	COLUMN ITEM 4=HISTORY

3.7 SICHERN UND LADEN VON KALKULATIONSTABELLEN

Die Daten im Speicher des Computers bleiben erhalten, auch wenn der Computer ausgeschaltet wird. Es können jedoch nur bis zu 10 Kalkulationstabellen im Speicher enthalten sein. Zum Umgang mit mehr als 10 Kalkulationstabellen oder zum Verarbeiten der Daten einer Kalkulationstabelle, die mit einem anderen PC-1425 erstellt wurde, muß entweder eine RAM-Karte eingesetzt oder eine Bandkassette verwendet werden. Dieser Abschnitt beschreibt das Sichern auf und das Laden von Kassette.

1. Sichern auf Kassette

Die jeweils angezeigte Kalkulationstabelle wird gesichert. Dabei ist es gleichgültig, welches Feld gerade angezeigt wird. Der Name der zu sichernden Datei kann aus bis zu 7 alphanumerischen Zeichen und Symbolen bestehen.

[SHIFT] [FILE]	1:SAVE 2:LOAD 3:VERIFY
1	FILE NAME=?
Dateiname [EXECUTE]	***EXECUTING***

Abbrechen des Sicherungsvorgangs
Wenn der Kassettenrekorder noch nicht mit der Aufnahme begonnen hat, wird durch Drücken der Taste [C·CE] zur vorherigen Anzeige zurückgekehrt.
Wenn der Kassettenrekorder bereits aufnimmt, kann das Sichern durch Drücken der Taste [BRK] abgebrochen werden.

2. Laden von Kassette

Nach Anzeige der Titel-Nummer der Kalkulationstabelle, die geladen werden soll, werden die auf Kassette gesicherten Daten für eine Kalkulationstabelle geladen. Der Name der Datei, in der die zu ladende Kalkulationstabelle gespeichert ist, kann aus bis zu 7 alphanumerischen Zeichen und Symbolen bestehen.

[SHIFT] [FILE]	1:SAVE 2:LOAD 3:VERIFY
2	FILE NAME=?
Dateiname [EXECUTE]	***EXECUTING***

Abbrechen des Ladevorgangs

Wenn der Kassettenrekorder noch nicht mit der Wiedergabe begonnen hat, wird durch Drücken der Taste [C·CE] zur vorherigen Anzeige zurückgekehrt.

Wenn der Kassettenrekorder bereits wiedergibt, kann das Laden durch Drücken der Taste [BRK] abgebrochen werden.

Hinweis: Wenn die Titel-Nummer einer Kalkulationstabelle, die bereits Daten enthält, spezifiziert wird, werden die vorhandenen Daten gelöscht und die neuen Daten von der Kassette eingelesen.

3. Überprüfen der auf Kassette gespeicherten Kalkulationstabelle

Mit dieser Funktion wird überprüft, ob die angezeigte Kalkulationstabelle mit der auf Kassette gesicherten übereinstimmt. Der Name der gesicherten Datei, die überprüft werden soll, kann aus bis zu 7 alphanumerischen Zeichen und Symbolen bestehen.

[SHIFT] [FILE]	1:SAVE 2:LOAD 3:VERIFY
3	FILE NAME=?
Dateiname [EXECUTE]	***EXECUTING***

Abbrechen der Überprüfung

Wenn der Kassettenrekorder noch nicht mit der Wiedergabe begonnen hat, wird durch Drücken der Taste [C·CE] zur vorherigen Anzeige zurückgekehrt.

Wenn der Kassettenrekorder bereits wiedergibt, kann das Überprüfen durch Drücken der Taste [BRK] abgebrochen werden.

Das vorherige Feld wird wieder angezeigt, wenn die Kalkulationstabelle im Computer mit der auf Kassette gesicherten übereinstimmt. Wenn nicht, wird "I/O ERROR" angezeigt. In diesem Fall die Überprüfung noch einmal durchführen oder die Kalkulationstabelle erneut auf Kassette sichern.

3.8 STATISTISCHE BERECHNUNGEN

Statistische Berechnungen mit Kalkulationstabellen-Daten verwenden die Summe der Quadrate. Aus diesem Grund tritt ein Fehler auf, wenn alle Datenwerte in der Kalkulationstabelle nicht innerhalb des Bereiches $0 \leqq |\text{Daten}| < 1 \times 10^{50}$ liegen. Weiterhin tritt ein Fehler auf, wenn eine Summe (z.B. Σx^2) größer oder gleich 10^{100} ist.

1. Statistische Berechnungen mit einer Variablen

Mit den Daten in der Kalkulationstabelle wollen wir die Summe und die Standardabweichung berechnen. Die statistischen Berechnungen werden nur durchgeführt, wenn ein Spalten-Name angezeigt wird, und dann mit den Daten dieser Spalte. Nehmen Sie die folgenden Tastenbetätigungen vor, während die Spalte ITEM 3=ENGLISH aus Tabelle 3-1 angezeigt wird:

	COLUMN ITEM 3=ENGLISH
[→X]	COLUMN ITEM 3 [>X]=ENGLI→
[EXECUTE]	STAT.CALCULATION OK?
[EXECUTE]	0.

Nach dieser Bedienung werden die Werte für n, Σx, Σx^2, $\bar{x}$, Sx, σx durch Drücken der entsprechenden Taste angezeigt.
Nicht eingegebene Datenwerte werden als fehlende Daten angesehen und bei den Berechnungen ausgelassen.
Mit den statistischen Berechnungen mit einer Variablen werden die folgenden statistischen Größen erhalten.

① n: Stichprobenumfang
② Σx: Summe der Stichprobenwerte
③ Σx^2: Summe der Quadrate der Stichprobenwerte
④ $\bar{x}$: Stichprobenmittelwert $\bar{x} = \dfrac{\Sigma x}{n}$
⑤ Sx: Standardabweichung, wenn der Grundgesamtheits-Parameter n-1 ist

$$Sx = \sqrt{\frac{\Sigma x^2 - n\bar{x}^2}{n-1}}$$

- Diese Formel dient zur Schätzung der Standardabweichung der Grundgesamtheit aufgrund der Daten, die der Grundgesamtheit als Stichproben entnommen wurden.

⑥ σx: Standardabweichung, wenn der Grundgesamtheits-Parameter n ist

$$\sigma x = \sqrt{\frac{\sum x^2 - n\bar{x}^2}{n}}$$

- Diese Formel dient zur Berechnung der Standardabweichung, wenn die gesamte Grundgesamtheit für die Stichprobendaten verwendet wird oder wenn die Stichprobe als Grundgesamtheit angesehen wird.

Beispiel 1: Führen Sie statistische Berechnungen mit einer Variablen für die Englisch-Punktzahlen in Tabelle 3-1 durch.

① n:	Stichprobenumfang	5.
② Σx:	Summe	308.
③ Σx^2:	Summe der Quadrate	21794.
④ $\bar{x}$:	Stichprobenmittelwert	61.6
⑤ Sx:	Stichproben-Standardabweichung	26.55748482
⑥ σx:	Grundgesamtheits-Standardabweichung	23.75373655

2. Statistische Berechnungen mit zwei Variablen

Mit den Daten in der Kalkulationstabelle wollen wir die Summe und die Standardabweichung berechnen. Die statistischen Berechnungen werden nur durchgeführt, wenn ein Spalten-Name angezeigt wird, und dann mit den Daten dieser Spalte. Nehmen Sie die folgenden Tastenbetätigungen vor, um die Englisch-Punktzahlen als x-Daten und die Mathematik-Punktzahlen als y-Daten zu verwenden:

	COLUMN ITEM 3=ENGLISH
[→X]	COLUMN ITEM [>X]=ENGLI→
[◄] [◄]	COLUMN ITEM 1=MATH
[SHIFT] [→Y]	COLUMN ITEM 1 [>Y]=MATH
[EXECUTE]	STAT.CALCULATION OK?
[EXECUTE]	Ø.

Nach dieser Bedienung werden statistischen Größen für zwei Variable durch Drücken der entsprechenden Taste angezeigt.
Wenn ein Wert eines geordneten Paares (x, y) nicht vorhanden ist, wird das geordnete Paar als fehlende Daten angesehen und bei den Berechnungen ausgelassen.

① n, Σx, Σx^2, $\bar{x}$, Sx, und σx gleich wie bei den Berechnungen mit einer Variablen
② Σy: Summe der y-Stichprobenwerte
③ Σy^2: Summe der Quadrate der y-Stichprobenwerte
④ Σxy: Summe der Produkte von x und y
⑤ $\bar{y}$: y-Stichprobenmitelwert $\bar{y}=\frac{\Sigma y}{n}$
⑥ Sy: Standardabweichung der y-Stichprobenwerte, wenn der Grundgesamtheits-Parameter n-1 ist $Sy=\sqrt{\frac{\Sigma y^2-n\bar{y}^2}{n-1}}$
⑦ σy: Standardabweichung der y-Stichprobenwerte, wenn der Grundgesamtheits-Parameter n ist $\sigma y=\sqrt{\frac{\Sigma y^2-n\bar{y}^2}{n}}$
⑧ a: Koeffizient der Primärregressionslinie $y=a+bx$

$$a=\bar{y}-b\bar{x}$$

⑨ b: Koeffizient der Primärregressionslinie $Y = a + bx$

$$b = \frac{Sxy}{Sxx}$$

⑩ r: Korrelationskoeffizient

$$r = \frac{Sxy}{\sqrt{Sxx \cdot Syy}}$$

⑪ x': Schätzwert (Schätzung des Wertes von x aus dem Wert von y)

$$x' = \frac{y - a}{b}$$

⑫ y': Schätzwert (Schätzung des Wertes von y aus dem Wert von x)

$y' = a + bx$

$$\left[\begin{array}{l} Sxx = \sum x^2 - \dfrac{(\sum x)^2}{n} \\ Syy = \sum y^2 - \dfrac{(\sum y)^2}{n} \\ Sxy = \sum xy - \dfrac{\sum x \cdot \sum y}{n} \end{array}\right]$$

Beispiel: Berechnung der linearen Regression

Berechnen Sie mit den oben beschriebenen Operationen die Koeffizienten a und b der Primärregressionslinie $y = a + bx$ und den Korrelationskoeffizienten r und schätzen Sie die Mathematik-Punktzahl eines Schülers, der eine Englisch-Punktzahl von 80 hat. Schätzen Sie weiterhin die Englisch-Punktzahl einer Schülers, der eine Mathematik-Punktzahl von 60 hat.

SHIFT a (Koeffizient a von $y = a + bx$)	−0.093293633
SHIFT b (Koeffizient b von $y = a + bx$)	0.910605416
SHIFT r (Korrelationskoeffizient)	0.836650247
80 y' (Schätzung des Wertes von y aus dem Wert von x)	72.75513966
60 x' (Schätzung des Wertes von x aus dem Wert von y)	65.99268198

Beispiel: Berechnung der Exponentialregression

Berechnen Sie mit den untenstehenden Daten die Koeffizienten a und b von $y = a \cdot e^{bx}$ sowie den Korrelationskoeffizienten r und schätzen Sie für x gleich 12 den Wert von y.

Nr.	A	B	C	D	E	F
x_i	2	7	9.2	4.3	5.1	8
y_i	0.6	4.02	8.3	1.21	2.7	5.1

Durchführung der Berechnung

Logarithmieren beiden Seiten ergibt

$\ln y = \ln a + bx$.....①

Substituieren von Y für $\ln y$ und A für $\ln a$ ändert die Gleichung (1) zu

$Y = A + bx$.....②

Jetzt können die Berechnungen durchgeführt werden. Da $Y = \ln y$ ist, kann y bei der Eingabe von Y logarithmiert werden.

Durchführung

[TABLE] [TABLE]
(Einstellung der Kalkulationstabellen-Betriebsart)

TABLE TITLE 2 = ?

EXAMPLE
(Titel eingeben)

TABLE TITLE 2 = EXAMPLE_

[DATA]

COLUMN ITEM 1 = ?

X [DATA] Y [DATA]

COLUMN ITEM 3 = ?

[DATA]

ROW ITEM 1 = ?

A [DATA] 2 [DATA]
[SHIFT] [ln] 0.6 [DATA]

ROW ITEM 2 = ?

B [DATA] 7 [DATA]
[SHIFT] [ln] 4.02 [DATA]

ROW ITEM 3 = ?

C [DATA] 9.2 [DATA]
[SHIFT] [ln] 8.3 [DATA]

ROW ITEM 4 = ?

D [DATA] 4.3 [DATA] [SHIFT] [ln] 1.21 [DATA]	ROW ITEM 5 = ?
E [DATA] 5.1 [DATA] [SHIFT] [ln] 2.7 [DATA]	ROW ITEM 6 = ?
F [DATA] 8 [DATA] [SHIFT] [ln] 5.1 [DATA]	ROW ITEM 7 = ?

Mit den Tasten [↑] und [►] zur nächsten Anzeige gehen.

	COLUMN ITEM 1 = X
[→X]	COLUMN ITEM 1 [>X] = X
[►]	COLUMN ITEM 2 = Y
[SHIFT] [→Y]	COLUMN ITEM 2 [>Y] = Y
[EXECUTE] [EXECUTE]	0.
[SHIFT] [a] [SHIFT] [e^x]	0.307632838 ($a = e^A$)
[SHIFT] [b]	0.361879612
[SHIFT] [r]	0.983506276
12 [y′] [SHIFT] [e^x]	23.65813575 (Schätzung: y ist etwa 23.66)

3.8 VERWENDUNG DER VERTEILUNGSFUNKTIONEN

Für die Verteilungsfunktionen können nur numerische Daten eingegeben werden. Ein Datenwert, der erst noch berechnet werden muß (beispielsweise ein Ausdruck wie 1/6), kann nicht eingegeben werden.
Diese Berechnungen können, abhängig von den verwendeten Variablen, einige Zeit benötigen.

Normalverteilung
Berechnen Sie den Punkt x für ein gegebenes oberes Wahrscheinlichkeitsintegral p ($0<p<1$).

Tastenbetätigung	Anzeige
[DISTRIBUTION] [N(p)]	p=?
0.05 [EXECUTE]	1.645

Berechnen Sie as obere Wahrscheinlichkeitsintegral p von x.

Tastenbetätigung	Anzeige
[DISTRIBUTION] [N(x)]	x=?
1.645 [EXECUTE]	0.05

t-Verteilung
Berechnen Sie den Punkt x für en gegebenen Freiheitsgrad n und ein gegebenes oberes Wahrscheinlichkeitsintegral p ($0<p<1$).

Tastenbetätigung	Anzeige
[DISTRIBUTION] [t(p)]	n=?
10 [EXECUTE]	p=?
0.01 [EXECUTE]	2.764

Berechnen Sie das obere Wahrscheinlichkeitsintegral p für ein gegebenes x der t-Verteilung mit Freiheitsgrad n (ganze Zahl im Bereich $1 \leqq n < 10^{10}$)

Tastenbetätigung	Anzeige
[DISTRIBUTION] [t(x)]	n=?
10 [EXECUTE]	x=?
−2.764 [EXECUTE]	0.99

Chi-Quadrat-Verteilung (χ^2)

Berechnen Sie den Punkt x für ein gegebenes oberes Wahrscheinlichkeitsintegral p der χ^2-Verteilung mit Freiheitsgrad n ($0<p\leqq 1$).

Tastenbetätigung	Anzeige
[DISTRIBUTION] [χ^2(p)]	n=?
3 [EXECUTE]	p=?
0.95 [EXECUTE]	0.3518

Berechnen Sie das obere Wahrscheinlichkeitsintegral p für ein gegebenes x der χ^2-Verteilung mit Freiheitsgrad n (ganze Zahl im Bereich $1\leqq n<10^{10}$)

Tastenbetätigung	Anzeige
[DISTRIBUTION] [$\chi^2(x)$]	n=?
3 [EXECUTE]	x=?
0.3518 [EXECUTE]	0.95

F-Verteilung

Berechnen Sie den Punkt x für ein gegebenes oberes Wahrscheinlichkeitsintegral p der F-Verteilung mit Freiheitsgrad n_1 und n_2 ($0<p\leqq 1$)

Tastenbetätigung	Anzeige
[DISTRIBUTION] [F(p)]	n_1=?
5 [EXECUTE]	n_2=?
10 [EXECUTE]	p=?
0.05 [EXECUTE]	3.326

Berechnen Sie das obere Wahrscheinlichkeitsintegral p für ein gegebenes x der F-Verteilung mit Freiheitsgrad n_1 und n_2 (ganze Zahlen im Bereich $1\leqq n_1$ und $n_2<10^{10}$)

Tastenbetätigung	Anzeige
[DISTRIBUTION] [F(x)]	n_1=?
5 [EXECUTE]	n_2=?
10 [EXECUTE]	x=?
3.326 [EXECUTE]	0.05

Binominal-Verteilung (B-Verteilung)

Berechnen Sie die Wahrscheinlichkeit, daß das Ereignis mit der Wahrscheinlichkeit p r-mal bei n Versuchen eintritt ($0<p<1$)

Beispiel: Bestimmen Sie mit einer unverfälschten Münze, die mit gleicher Häufigkeit mit Zahl und Wappen nach oben fällt, die Wahrscheinlichkeit von dreimal Zahl bei 7 Würfen.

Tastenbetätigung	Anzeige
[DISTRIBUTION] [J/B]	p=?
0.5 [EXECUTE]	n=?
7 [EXECUTE]	r=?
3 [EXECUTE]	CUMULATION YES(1) NO(2)?
2	Ø.273438

Durch Eingabe von "1" (YES) für die kumulative Wahrscheinlichkeit erhalten Sie die Wahrscheinlichkeit für dreimal oder häufiger Zahl.

Poisson-Verteilung (P-Verteilung)

Berechnen Sie die Wahrscheinlichkeit, daß das Ereignis mit dem Mittelwert m x-mal eintritt (große Grundgesamtheit)

Beispiel: In einer Stadt gibt es im Durchschnitt 1,8 Unfälle pro Tag. Berechnen Sie die Wahrscheinlichkeit eines Tages ohne Unfälle.

Tasteneingabe	Anzeige
[DISTRIBUTION] [K/P]	m=?
1.8 [EXECUTE]	x=?
0 [EXECUTE]	CUMULATION YES(1) NO(2)?
2	Ø.165299

3.10 DRUCKEN DER KALKULATIONSTABELLE

1. Drucken von Kalkulationstabellen

Sie können die folgenden Drucker einsetzen: CE-126P, CE-140P oder CE-515P.
Wir wollen die früher eingegebene Kalkulationstabelle ausdrucken.

	TABLE TITLE 1=SCORES
SHIFT PRINT	PLEASE PUSH ENTER
EXECUTE	***EXECUTING***

Tabelle 3-3 wird in Fließkomma-Darstellung ausgedruckt, siehe unten.

```
SCORES
001:BOND
002:JONSON
003:THOMPSON
004:ADAMS
005:YOUNG

MATH
001:          20.
002:          39.
003:          51.
004:          80.
005:          90.
COLUMN TOTAL
             280.
COLUMN MEAN
              56.

SCIENCE
001:          30.
002:          41.
003:          22.
004:          90.
005:          80.
COLUMN TOTAL
             263.
COLUMN MEAN
              52.6

ENGLISH
001:          25.
002:          63.
003:          60.
004:          60.
005:         100.
COLUMN TOTAL
             308.
COLUMN MEAN
              61.6

HISTORY
001:          69.
002:          38.
003:          70.
004:          50.
005:          70.
COLUMN TOTAL
             297.
COLUMN MEAN
              59.4

ROW TOTAL
001:         144.
002:         181.
003:         203.
004:         280.
005:         340.
TOTAL
            1148.

ROW MEAN
001:          36.
002:          45.25
003:          50.75
004:          70.
005:          85.
```

Drucken nur einer Spalte
Wenn ein eingegebener Datenwert angezeigt wird, wird nur die Spalte ausgedruckt, die das Feld mit diesem Datenwert enthält.

2. Drucker-Wahl

Obwohl viele verschiedene Drucker angeschlossen werden können, wird der aktive Drucker automatisch nach den folgenden Situationen gewählt.

1. Wenn der CE-126P angeschlossen ist, druckt er immer unabhängig davon, ob noch ein anderer Drucker angeschlossen ist.
2. Der CE-515P druckt nur, wenn der CE-126P und CE- 140P nicht angeschlossen sind.
   Auch wenn der Schalter des CE-140P für SIO eingestellt ist, druckt der CE-140P, falls angeschlossen.

Hinweis: Zum Drucken mit dem CE-140P oder dem CE-515P wird CSIZE (Zeichengröße-Zuweisung) nicht eingestellt oder geändert. Daher können Situationen eintreten, in denen einwandfreies Drucken nicht möglich ist. In solchen Fällen den Drucker ausschalten und dann vor dem Drucken wieder einschalten.

3.11 VERWENDUNG DER DATEN MIT BASIC

Die Daten und statistischen Größen, die bei den Kalkulationstabellen-Berechnungen gespeichert werden, können auch aus einem BASIC-Programm, das sie als vorbestimmte Variable gespeichert hat, verwendet werden.
Andererseits können auch mit einem BASIC-Programm Daten in diese Variable gespeichert und dann in Kalkulationstabellen-Berechnungen verwendet werden.
Diese Variablen werden bei Ausführung des RUN-, NEW- oder CLEAR-Befehls in BASIC nicht gelöscht. Zum Löschen dieser Variablen muß der Befehl ERASE verwendet werden.

Kalkulationstabellen-Daten

Die verschiedenen in der Kalkulationstabelle verwendeten Daten werden in Matrizen mit den unten aufgeführten Namen gespeichert.

	Tabelle 1	Tabelle 2	— — — —	Tabelle 10
Kalkulationstabellen-Titel und Spalten-Name	T0$ (n)	T1$ (n)	— — — —	T9$ (n)
Zeilen-Namen	D0$ (m)	D1$ (m)	— — — —	D9$ (m)
Daten	D0 (a,b)	D1 (a,b)	— — — —	D9 (a,b)

Beispiel von Kalkulationstabellen-Daten (Tabelle 2-1)

SCORES

	MATH	SCIENCE	ENGLISH	HISTORY	ROW TOTAL	ROW MEAN
ADAMS	80.	90.	60.	50.	280.	70.
BOND	20.	30.	25.	69.	144.	36.
JOHNSON	39.	41.	63.	38.	181.	45.25
THOMPSON	51.	22.	60.	70.	203.	50.75
YOUNG	90.	80.	100.	70.	340.	80.
COLUMN TOTAL	280.	263.	308.	297.	1148.	— — — —
COLUMN MEAN	56.	52.6	61.6	59.4	— — — —	— — — —

Diese Daten für Titel 1 werden unter den entsprechenden Variablennamen in der untenstehenden Tabelle gespeichert.

BASIC-Variable

T0$ (4)

	T0$(0)	**T0$(1)**	**T0$(2)**	**T0$(3)**		
D0$(0)	D0(0,0)	D0(0,1)	D0(0,2)	D0(0,3)	D0(0,4)	D0(0,5)
D0$(1)	D0(1,0)	D0(1,1)	D0(1,2)	D0(1,3)	D0(1,4)	D0(1,5)
D0$(2)	D0(2,0)	D0(2,1)	D0(2,2)	D0(2,3)	D0(2,4)	D0(2,5)
D0$(3)	D0(3,0)	D0(3,1)	D0(3,2)	D0(3,3)	D0(3,4)	D0(3,5)
D0$(4)	D0(4,0)	D0(4,1)	D0(4,2)	D0(4,3)	D0(4,4)	D0(4,5)
	D0(5,0)	D0(5,1)	D0(5,2)	D0(5,3)	D0(5,4)	D0(5,5)
	D0(6,0)	D0(6,1)	D0(6,2)	D0(6,3)	D0(6,4)	D0(6,5)

Hinweise:

1. Bei Dateneingabe in Kalkulationstabellen-Berechnungen wird automatisch der mit dem DIM-Befehl eingestellte Bereich zugewiesen.
2. Selbst wenn die Summen und Mittelwerte nicht berechnet werden, werden ihre Variablenbereiche zugewiesen und auf 0 gesetzt.
3. Wenn eine Summe und ein Mittelwert berechnet wird, werden die Bereiche für D0(5,5) und D0(6,4) gleich 0 und wird der Bereich für D0(6,5) gleich 1.
4. Zur Zuweisung des Bereichs einer Kalkulationstabelle in BASIC dient DIM T0$( )*32, D0$( )*32, D0(,) und 32 für die Anzahl der Zeichen. Außerdem muß Bereich für die Summen und Mittelwerte zugewiesen werden. Die Daten für Kalkulationstabellen-Berechnungen können nur dann gemeinsam genutzt werden, wenn diese 3 Feldvariablen gleichzeitig vorhanden sind. Sie werden durch Ausführung der Befehle RUN, CLEAR und NEW nicht gelöscht. Zum Löschen dient der Befehl ERASE.
5. Nach Speicherung der Daten in BASIC werden die Summen und Mittelwerte als vorhanden angenommen, wenn der Bereich für D0(6,5) nicht Null ist, und die gespeicherten Daten werden angezeigt. In diesem Fall können die Werte der Summen und Mittelwerte falsch sein.
   Daher sollte eine Neuberechnung mit den Tasten [SHIFT] [TOTAL] in der Kalkulationstabellen-Betriebsart durchgeführt werden.
6. Der Wert von fehlenden Daten wird $9.999999999 \times 10^{99}$, und der Wert von nicht verfügbaren Daten wird $-9.999999999 \times 10^{99}$. Wenn ein in BASIC eingegebener numerischer Wert 10^{99} überschreitet, wird er als fehlender Datenwert verarbeitet, und wenn er kleiner als -1×10^{99} ist, wird er als nicht verfügbarer Datenwert verarbeitet.

Statistische Werte

Die verschiedenen statistischen Werte werden in den folgenden Variablen gespeichert.

Statistischer Wert		Variable
Stichprobenumfang	n	Z
Summe von x	Σx	Y
Summe der x-Quadrate	Σx^2	X
Summe der Produkte von x und y	Σxy	W
Summe von y	Σy	V
Summe der y-Quadrate	Σy^2	U
Mittelwert von x	$\overline{x}$	T
Stichproben-Standardabweichung von x	Sx	S
Grundgesamtheit-Standardabweichung von x	σx	R
Mittelwert von y	$\overline{y}$	Q
Stichproben-Standardabweichung von y	Sy	P
Grundgesamtheit-Standardabweichung von y	σy	O
Korrelationskoeffizient von x und y	r	N
y-Achsenabschnitt der Primärregressionslinie	a	M
Steigung der Primärregressionslinie	b	L

Diese Datenwerte werden durch Ausführung der Befehle CLEAR und NEW gelöscht.

KAPITEL 4
VERWENDUNG ALS WISSENSCHAFTLICHER RECHNER

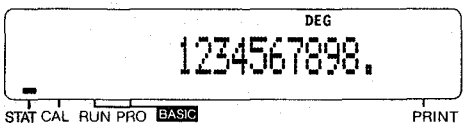
4.1 BETRIEBSHINWEISE

Die folgenden Absätze enthalten einige wichtige Hinweise, die sich auf die CAL-Betriebsart des Rechners beziehen. Die in KAPITEL 1 und 2 enthaltenen allgemeinen Betriebshinweise haben darüber hinaus Gültigkeit.
Das Umschalten des Rechners in die CAL-Betriebsart ([STAT] -Taste drücken) wurde bereits in KAPITEL 2 aufgezeigt, so daß an dieser Stelle auf eine nähere Erläuterung dieser Punkte verzichtet werden kann.

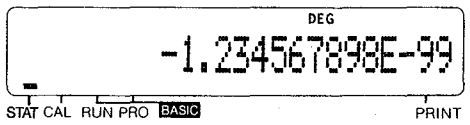
Zweite Funktionsebene der Tastatur; Merkmale; Editier- und Korrekturmöglichkeiten

Wie aus der nachfolgenden Abbildung ersichtlich, weisen nahezu alle Tasten des Tastenfelds der CAL-Betriebsart eine zweite Funktionsebene auf. In den Erläuterungen und Rechenbeispielen ist die Auswahl der zweiten Tastenfunktion aus der Verwendung der [SHIFT] -Taste ersichtlich.
Bei näherer Betrachtung des Tastenfelds der CAL-Batriebsart wird bereits ein wichtiges Merkmal dieses Computers deutlich. Es steht eine große Anzahl von Tastenfunktionen zur Verfügung, deren Verwendung zu einer wesentlichen Arbeitserleichterung und Arbeitsvereinfachung beiträgt.

Anzeige/Display-Format und Symbole.



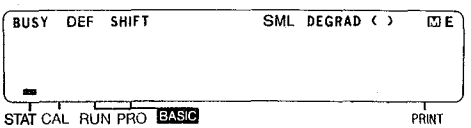
Normales Displayformat



Exponentielles Displayformat

Mantisse (12 Stellen) Exponent (4 Stellen)

Die Anzeige des Computers umfaßt 24 Stellen, von denen 16 Stellen zur Anzeige von Zahlen dienen. In der Betriebsart "CAL" werden die Rechenergebnisse soweit möglich in Fließkommaschreibweise dargestellt. Ist das Rechenergebis kleiner 0,000000001 oder größer 9999999999 (größer −0,000000001) oder kleiner −9999999999), erfolgt die Anzeige in wissenschaftlicher Schreibweise. Dabei wird die Mantisse 12 stellig (einschließlich Vorzeichen und Komma) und der Exponent 4 stellig (einschließlich Symbol und Vorzeichen) angezeigt.



Die oben abgebildeten Symbole und Markierungen, die nachfolgend aufgelistet sind und deren Bedeutung erläutert wird, zeigen die Betriebsart und den Status des Computers an.

SHIFT Das Wort "SHIFT" wird angezeigt, wenn die [SHIFT]-Taste gedrückt wurde und damit die zweite Funktionsebene der Tasten angewählt ist.

DEF Leuchtet, wenn die Taste [DEF] gedrückt wird, und zeigt an, daß in der STAT-Betriebsart eine Verteilungsfunktion bzw. daß in der RUN-Betriebsart die Ausführung eines mit Label versehenen Programms gewählt wurde.

SML Leuchtet durch Drücken der Taste [SML]. Dann werden Buchstaben als Kleinbuchstaben eingegeben.

DEG
RAD
GRAD

Diese drei Wörter können durch wiederholtes Betätigen der Tasten [SHIFT] und [DRG] angewählt werden. Die Wörter geben an, in welcher Winkeleinheit gerechnet wird.

DEG : (Grad; °, ', ")
RAD : (Radiant; rad)
GRAD : (Neugrad; gon)
Gestreckter Winkel = 180° = π rad = 200 gon

( )

Das Klammersymbol erscheint, sobald beim Eintippen einer Berechnungsformel die [(] -Taste verwendet wird.

M

Dieses Symbol zeigt an, daß sich ein numerischer Wert, der ungleich Null ist, im Speicher befindet und der Speicher somit belegt ist.

E

Das Symbol "E" in der rechten oberen Ecke des Anzeigefelds zeigt an, daß eine Bereichsüberschreitung aufgetreten ist oder eine falsche Anweisung gegeben wurde. Der Fehlerzustand kann durch das Betätigen der [C·CE] -Taste beseitigt werden.

CAL

Die Markierung (–) über der Beschriftung "CAL" gibt an, daß sich der Rechner in der Betriebsart "CAL" befindet.

STAT

Die Markierung (–) über der Beschriftung "STAT" im rechten unteren Bereich des Anzeigefelds gibt an, daß sich der Rechner in der Betriebsart "Statistische Berechnungen" befindet. Die Betriebsart wird durch Drücken der Taste [STAT] angewählt und durch nochmaliges Drücken wieder ausgeschaltet.

RUN

Der Strich (–) über RUN unten links auf dem Display zeigt an, daß der Computer in der RUN-Betriebsart (zum Ausführen von Programmen) ist.

PRO

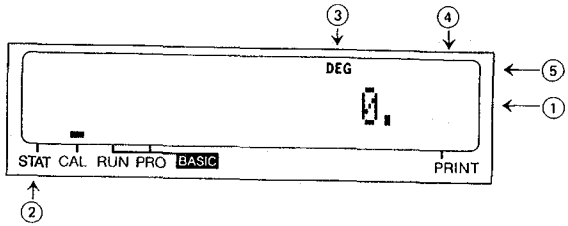
Der Strich (–) über PRO unten links auf dem Display bedeutet, daß der Computer in der PRO- Betriebsart (zum Schreiben von Programmen) ist.

BUSY

Diese Anzeige erscheint, während der Computer eine arithmetische Berechnung durchführt.

Grundeinstellung

Bevor mit den Berechnungen begonnen wird, sollte die Grundeinstellung des Rechners überprüft werden. Dazu den POWER-Schalter auf ON schalten oder, falls der Rechner automatisch abgeschaltet hat, die [ON BRK] -Taste drücken. Anschließend die [STAT] -Taste (grün) drücken, um so die Betriebsart "CAL" anzuwählen und zweimal die [C·CE] -Taste betätigen. Das Anzeigefeld sollte nun, entsprechend der nachfolgenden Abbildung, nur die Anfangsinformationen enthalten.



Falls dies nicht der Fall ist, die folgenden Kurzanweisungen lesen und die erforderlichen Maßnahmen durchführen.

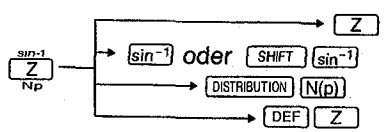
① Es wird mehr als eine Null angezeigt (z.b. **0.00**)
Die Anzahl der Stellen hinter dem Komma wird festgelegt. Die Tabulator-Einstellung wird durch Aus- und erneutes Einschalten des Rechners gelöscht. Der Computer befindet sich danach im normalen Displaymodus.

② Über der Beschriftung "STAT" wird eine Markierung (−) angezeigt:
Die Betriebsart "Statistische Berechnungen" ist angewählt.
Durch Drücken der Tasten [STAT] oder [BASIC] die Betriebsart aufheben.

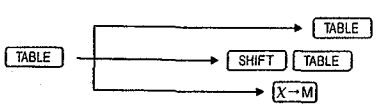
③ Statt DEG wird RAD oder GRAD angezeigt:
Die Winkeleinheit ist in Radiant (rad) oder Neugrad (gon) vorgewählt. Durch Drücken der Taste [SHIFT] [DRG] die Winkeleinheit auf DEG einstellen.

④ Das Symbol M wird angezeigt:
Der Speicher ist mit einem numerischen Wert belegt. Löschen des Speichers durch Drücken der Tasten [C·CE] [X→M].

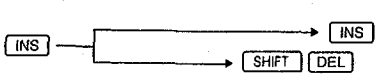
⑤ Alle im oberen Bereich des Anzeigefelds dargestellten Symbole, mit Ausnahme der in Punkt ③ und ④ beschriebenen, können durch Drücken der [C·CE] -Taste gelöscht werden.



: spezifiziert das Zeichen "z"
: spezifiziert $\sin^{-1}$
: spezifiziert N(p) im STAT-Modus
: spezifiziert Ausführung des Programms "Z" im RUN-Modus



{ spezifiziert Kalkulationstabellen-Berechnungen im STAT-Modus
: Speicherung im CAL-Modus



: spezifiziert Einfügung
: spezifiziert Löschung

Löschen der Eingabe oder einer Fehlermeldung

Durch einmaliges Drücken der roten [C·CE] -Taste wird die letzte Eingabe bzw. der Displayinhalt gelöscht. Ein zweimaliges Drücken der [C·CE] -Taste löscht auch den zuerst eingegebenen Wert bzw. das Ergebnis der vorangegangenen Berechnung.
Der Speicherinhalt [M] wird dadurch nicht gelöscht.

123 [+] 456	456.	6 [×] 2 [+]	12.
[C·CE]	0.	[C·CE]	0.
789 [=]	912.	6 [÷] 2 [+]	3.
(123 + 789 = 912)		5 [=]	8.

Die [C·CE] -Taste dient darüber hinaus zum Löschen der Fehlermeldung (E).

5 [÷] 0 [=]	0. E	← Fehler (Error)-Symbol
[C·CE]	0.	

Auf-/Abrundung bzw. Festlegung der Nachkommastellen

Die Auf- oder Abrundung des Rechenergebnisses erfolgt in der Regel in der 11. Stelle. Da das Display 10 Stellen aufweist, ist die letzte Stelle des angezeigten Rechenergebnisses nicht auf- oder abgerundet. Bei benützung der Tasten [=] oder [M+] wird der angezeigte, nicht gerundete Wert im Falle weiterer Berechnungen verwendet.
Beispiel:

1 [÷] 3 [×] 3 [=]	1.
1 [÷] 3 [=]	0.333333333
[×] 3 [=]	0.999999999

Die Anzahl der Nachkommastellen und damit die Rundung kann frei festgelegt werden. Dazu dient die [TAB] -Taste in Verbindung mit den Tasten (0) bis (9).

[SHIFT] [TAB] [0] → Keine Nachkommastellen. Entsprechend der 1. Nachkommastelle wird auf- oder abgerundet.

[SHIFT] [TAB] [1] → Eine Nachkommastelle. Entsprechend der 2. Nachkommastelle wird auf- oder abgerundet.

[SHIFT] [TAB] [9] → Neun Nachkommastellen. Entsprechend der 10. Nachkommastelle wird auf- oder abgerundet.

Beispiel: [SHIFT] [TAB] [9]

[•] [5] [÷] [9] [=] → 0.055555556 (FIX-Betriebsart)
Neun Nachkommastellen. Die 9. Nachkommastelle wurde aufgerundet.

[F↔E] → 5.555555556E–02 (SCI-Betriebsart)
Neun Nachkommastellen. Die 9. Nachkommastelle der Mantisse wurde aufgerundet.

[SHIFT] [TAB] [3] → 5.556E–02 (SCI-Betriebsart)
Drei Nachkommastellen. Die 3. Nachkommastelle der Mantisse wurde aufgerundet.

[F↔E] → 0.055555555 (Normale Betriebsart)
Der Wert ist im Rechner in der Form von $5.55555555555 \times 10^{-2}$ gespeichert.
Die Rundung in der 9. Nachkommastelle der Mantisse ergibt $5.555555556 \times 10^{-2}$.
Da das Display 10 Stellen anzeigt, ist die Aufrundung in der Ergebnisanzeige nicht enthalten.

[F↔E] → 0.056 (FIX-Betriebsart)

Wissenschaftliche Schreibweise

Die Displayanzeige erfolgt, solange die Anzeigekapazität nicht unter- oder überschritten wird, in Festkommaschreibweise. Durch Drücken der [F↔E] -Taste kann jedoch auf einfache Weise zwischen "Festkomma-" und "Wissenschaftlicher Schreibweise" gewählt werden.
Soll oder muß ein Wert in wissenschaftlicher Schreibweise eingegeben werden, erfolgt zur Eingabe des Exponenten mit den Tasten [SHIFT] [E] . Das Vorzeichen des Exponenten läßt sich mit der [+/–] -Taste bestimmen.

Beispiel: 23 [X] 1000 [=] → 23000.

[SHIFT] [F↔E] → 2.3E 04

[SHIFT] [F↔E] → 23000.

[C·CE] 4 [SHIFT] [E] 3 → 4.E 03

(4×10^3)

[=]	4000.
1.23 [+/-]	-1.23
[SHIFT] [E] 5 [+/-]	-1.23E-05
	(-1.23×10^{-5})
[=]	-0.0000123

4.2 NORMALE BERECHNUNGEN

Die nachfolgenden Beispiele zeigen die Bedienung des Rechners bzw. die Tastenfolge für die Grundrechenfunktionen.
Bevor damit begonnen wird, die Beispiele nachzuvollziehen, sollte, um eine gemeinsame Ausgangsbasis sicherzustellen, eine kurze Überprüfung der Grundeinstellung des Computers erfolgen.

POWER-Schalter auf ON schalten
(Falls der Rechner automatisch abgeschaltet hat, zum Einschalten die [ON BRK] -Taste drücken)
[STAT] -Taste (grün) drücken
[C·CE] -Taste (rot) zweimal drücken

Das Anzeigefeld sollte nun nur die Anfangsinformationen enthalten. Ist dies nicht der Fall, die Grundeinstellung des Computers gemäß Seite 59 durchführen.

Addition und Subtraktion

Aufgabe: Rechnen von 123 + 654

Tastenfolge	Anzeige
[1] [2] [3]	123.
[+]	123.
[6] [5] [4]	654.
[=]	777.
	(123 + 654 = 777)

Aufgabe: 12 + 45,6 – 32,1 + 789 – 741 + 213
Tastenfolge: 12 [+] 45.6 [-] 32.1 [+] 789 [-] 741 [+] 213 [=]
Ergebnis: 286.5

Multiplikation und Division

Aufgabe: 841 x 586 / 0,12
Tastenfolge: 841 [X] 586 [÷] .12 [=]
Ergebnis: 4106883.333

Aufgabe: 427 + 54 x 32 / 7 – 39 x 2
Tastenfolge: 427 [+] 54 [X] 32 [÷] 7 [–] 39 [X] 2 [=]
Ergebnis: 595.8571429

Anmerkung: Multiplikation und Division haben Vorrang vor Addition und Subtraktion. Die Vorrangordnung wird beachtet. Der Computer führt intern zuerst die Multiplikation und Division durch.

Multiplikationen mit einer Konstanten

Der zuerst eingegebene Wert kann als Konstante verwendet werden.
Aufgabe: 5 x 3 = ?, 10 x 3 = ? usw.
Tastenfolge: 3 [X] 5 [=]
Ergebnis: 15
Tastenfolge: 10 [=]
Ergebnis: 30

Divisionen mit einer Konstanten

Der als zweiter eingegebene Wert kann als Konstante verwendet werden.
Aufgabe: 15 / 3 = ?, 30 / 3 = ?, usw.
Tastenfolge: 15 [÷] 3 [=]
Ergebnis: 5
Tastenfolge: 30 [=]
Ergebnis: 10

Hinweis: Bei Kettenrechnungen bleiben entsprechend der Vorrangordnung die letzte Recheninstruktion und der jeweilige numerische Wert erhalten und können damit für weitere Berechnungen bzw. als Konstanten verwendet werden.

Ausgeführter Rechenvorgang	Konstante	
a + b x c =	+bc	(Konstanten-Addition)
a x b ÷ c =	÷c	(Konstanten-Division)
a ÷ b x c =	$\frac{a}{b}$ x	(Konstanten-Multiplikation)
a x b – c =	–c	(Konstanten-Subtraktion)

Potenz- und Wurzelfunktion

Aufgabe: 20^2
Tastenfolge: 20 [x^2]
Ergebnis: 400

Aufgabe: 3^3
Tastenfolge: 3 [y^x] 3 [=]
Ergebnis: 27

Aufgabe: 3^4
Tastenfolge: 3 [y^x] 4 [=]
Ergebnis: 81

Aufgabe: $\sqrt{25}$
Tastenfolge: 25 [$\sqrt{}$]
Ergebnis: 5

Aufgabe: 4. Wurzel von 81
Tastenfolge: 81 [$\sqrt[x]{y}$] 4 [=]
Ergebnis: 3

Prozentrechnung

Aufgabe: 45% von 2780 $\left(2{,}780 \times \frac{45}{100}\right)$
Tastenfolge: 2780 [×] 45 [%] [=]
Ergebnis: 1251

Aufgabe: 30% Abschlag von 200 $\left(200 - 200 \times \frac{30}{100}\right)$
Tastenfolge: 200 [−] 30 [%] [=]
Ergebnis: 140

Speicherrechnung (nur in der CAL-Betriebsart)

Der zur Verfügung stehende Speicher wird über die blau beschrifteten Tasten [x→M] [RM] [M+] angesprochen. Bevor mit dem Speicher gearbeitet wird, muß dessen Inhalt ggf. erst gelöscht werden. Der Speicherinhalt bleibt auch bei ausgeschaltetem Rechner erhalten. Die Belegung des Speichers zeigt das Symbol [M] in der rechten oberen Ecke des Displays an. Zum Löschen des Speichers die Tasten [C·CE] und [x→M] drücken.

Das Addieren eines Rechenergebnisses oder eines eingegebenen Werts zum Speicherinhalt erfolgt durch Drücken der [M+] -Taste. Das Drücken der [RM] -Taste bringt den Speicherinhalt am Display zur Anzeige.

Tastenfolge	Anzeige
12 [+] 5 [M+]	17
[C·CE]	0
[RM]	17
8 [÷] 2 [M+]	4
[RM]	21 (Speicherinhalt)

Soll ein Rechenergebnis oder ein eingegebener Wert vom Speicherinhalt subtrahiert werden, muß vor dem Betätigen der [M+] -Taste durch Drücken der [+/−] -Taste das Vorzeichen gewechselt werden.

Tastenfolge	Anzeige
2 [+] 5 [=] [+/−] [M+]	−7
[C·CE]	0
[RM]	14

Das Drücken der [x→M] -Taste überträgt den im Display angezeigten Wert in den Speicher. Der Speicherinhalt wird überschrieben.

Tastenfolge	Anzeige
12 [×] 2 [=] [x→M]	24
[C·CE]	0
[RM]	24

Vorrangordnung und Verwendung der "Klammern"-Tasten

Die Verwendung der Tasten [(] und [)] ist dann zwingend notwendig, wenn eine Serie von Einzelberechnungen zu einem Rechenvorgang zusammengefaßt werden soll und dabei die Vorrangordnung der Operatoren außer Kraft gesetzt werden muß.

Sobald die [(] -Taste gedrückt wird, erscheint in der rechten oberen Ecke des Anzeigefelds das ( )-Symbol, welches so lange sichtbar bleibt, bis alle geöffneten Klammern wieder geschlossen wurden. Das Schließen der Klammer(n) über die [)] -Taste kann unterbleiben, wenn unmittelbar folgend die Tasten [=] oder [M+] zu drücken sind, da diese die gleiche Funktion erfüllen.

Die Berechnungen in der Klammer haben Priorität vor allen anderen Berechnungen. Die Klammerfunktion kann in einer Ebene bis zu fünfzehnmal verwendet werden. Als erstes werden die Berechnungen der innersten Klammer ausgeführt.

Aufgabe: $12 + 42 \div (8 - 6)$

Tastenfolge: 12 [+] 42 [÷] [SHIFT] [(] 8 [−] 6 [SHIFT] [)] [=]

Ergebnis: 33

Aufgabe: $126 \div [(3 + 4) \times (3 - 1)]$

Tastenfolge: 126 [÷] [SHIFT] [(] [SHIFT] [(] 3 [+] 4 [SHIFT] [)] [×]
[SHIFT] [(] 3 [−] 1 [SHIFT] [)] [SHIFT] [)] [=]

Ergebnis: 9

Anmerkung: Das Schließen der beiden Klammern unmittelbar vor der [=]-Taste (oder [M+]-Taste) kann unterbleiben.

Vorrangordnung

Da der Computer die Vorrangordnung der einzelnen Berechnungen beachtet, ist es möglich, immer dann mathematische Formeln direkt (d.h. ohne daß Klammern eingefügt oder Formelumstellungen vorgenommen werden müssen) einzugeben, wenn es die Vorrangordnung nicht erfordert. Die Vorrangordnung bzw. die Reihenfolge, nach der die Berechnungen in einer längeren Rechenformel ausgeführt werden, ist wie folgt:

(1) Funktionen wie sin oder $\cos^{-1}$
(2) y^x, $\sqrt[x]{y}$, nPr, nCr
(3) x, ÷
(4) +, −
(5) =, M+

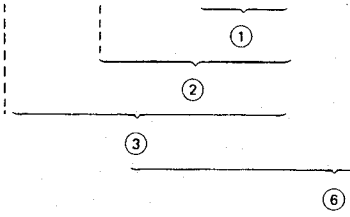
(Berechnungen, die gleiche Priorität besitzen, werden der Reihe nach ausgeführt)

Beispiel: Tastenfolge und Reihenfolge der Berechnungen bei

$5 + 2 \times \sin 30 + 24 \times 5^3 = 3006$

Winkeleinheit DEG

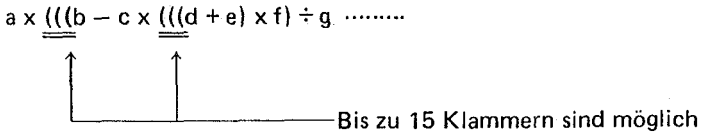
Tastenfolge: 5 [+] 2 [×] 30 [sin] [+] 24 [×] 5 [y^x] 3 [=]



Die Nummern ① ~ ⑥ geben die Reihenfolge der Berechnungen an.

Um die Berechnungen gemäß der Vorrangordnung bzw. der Verwendung von Klammern ausführen zu können, ist der Computer mit zusätzlichen Speichern ausgestattet. Diese ermöglichen es, daß mathematische Formeln, die bis zu acht Berechnungsebenen erforderlich machen, direkt eingegeben und berechnet werden können.

Die Klammerfunktion, welche ebenfalls eine Berechnungsebene ergibt, kann, solange keine weiteren Ebenen notwendig sind, bis zu fünfzehnmal verwendet werden.



Funktionen mit nur einen Variablen wie (x^2, n! usw.) werden sofort nach dem Drükken der Taste berechnet und benötigen somit keine Ebene bzw. keinen Speicher.

Beispiel: Berechnungsebenen bzw. Zwischenspeicher (ohne Klammerfunktion)

1 [+] 2 [=]	(3)	1 Ebene
1 [+] 2 [×] 3 [=]	(7)	2 Ebenen
1 [+] 2 [×] 3 [y^(1/x)] 4 [=]	(163)	3 Ebenen
1 [+] 2 [×] 3 [y^(1/x)] 4 [÷] 5 [=]	(33,4)	3 und anschl. 2 Ebenen

Mit Drücken der [y^(1/x)] -Taste sind 3 Ebenen erreicht. Nach Drücken der [÷] -Taste werden die Berechnungen "y^x" mit höherer Priorität und "x" mit gleicher Priorität als als "/" ausgeführt. Es bleiben somit nach Drücken der [÷] -Taste noch 2 Ebenen erhalten. Nach Drücken der [=] -Taste wird zuerst "/" (höhere Priorität) und anschließend "+" ausgeführt.

Beispiel: Berechnungsebenen bzw. Zwischenspeicher
(mit Klammerfunktion)

1 [+] 2 [×] 3 [y^(x)] [SHIFT] [(] 4 [÷] 5 [)] [=]

① ② ③ ④

(5,81644937) 4 Ebenen

Die Berechnung in der Klammer wird zuerst ausgeführt. Anschließend folgt "y^x", dann "x" und zuletzt "+".

1 [+] 2 [×] [SHIFT] [(] 3 [−] 4 [÷] 5 [SHIFT] [)] [=]

① ② ③ ④

(5,4) 4 und anschl. 3 Ebenen

Nach Drücken der [(] -Taste wird zuerst "/", dann "−" ausgeführt. Auf das Drücken der [=] -Taste folgt "x" und zuletzt "+".

4.3 WISSENSCHAFTLICHE BERECHNUNGEN

Bei der Durchführung von Berechnungen, die Winkelwerte enthalten, muß auf die Winkeleinheit, die verwendet wird, geachtet werden. Das Vorwählen der Winkeleinheit erfolgt durch Drücken der Tasten [SHIFT] und [DRG] . Die gewählte Winkeleinheit wird im Display angezeigt.

Trigonometrische und inverse trinometrische Funktionen (Arcusfunktionen)

Aufgabe: $\sin 30^\circ + \cos 40^\circ$
Tastenfolge: Winkeleinheit auf DEG
30 [sin] + 40 [cos] =
Ergebnis: 1,266044443

Aufgabe: $\cos 0{,}25\pi$
Tastenfolge: Winkeleinheit auf RAD
.25 [×] [P] [=] [cos]
Ergebnis: 0,707106781
Hinweis: π kann durch Drücken der Taste [P] berechnet werden.

Aufgabe: $\arcsin 0{,}5$
Tastenfolge: Winkeleinheit auf DEG
.5 [sin⁻¹]
Ergebnis: 30

Aufgabe: $\arccos -1$
Tastenfolge: Winkeleinheit auf RAD
1 [+/−] [cos⁻¹]
Ergebnis: 3,141592654 (Wert π)

Das Ergebnis einer inversen trigonometrischen Funktion kann nur in folgendem Bereich liegen:

$\theta = \sin^{-1} x$, $\theta = \tan^{-1} x$	$\theta = \cos^{-1} x$
DEG: $-90 \leqq \theta \leqq 90$ [°]	DEG: $0 \leqq \theta \leqq 180$ [°]
RAD: $-\frac{\pi}{2} \leqq \theta \leqq \frac{\pi}{2}$ [rad]	RAD: $0 \leqq \theta \leqq \pi$ [rad]
GRAD: $-100 \leqq \theta \leqq 100$ [g]	GRAD: $0 \leqq \theta \leqq 200$ [g]

Logarithmische Funktionen

Aufgabe: ln 21, log 173

Tastenfolge: Natürlicher Logarithmus 21 [ln]

Ergebnis: 3,044522438

Allgemeiner Logarithmus 173 [log]

Ergebnis: 2,238046103

Exponentialfunktionen

Aufgabe: $e^{3.0445}$ $10^{2.238}$

Tastenfolge: Exponentialfunktion (e) 3.0445 [e^x]

Ergebnis: 20,99952881 (siehe ln 21)

Allgem. Exponentialfunktion 2.238 [10^x]

Ergebnis: 172,9816359 (siehe log 173)

Fakultät

Aufgabe: 69! (n! = 1 x 2 x 3 x . . . n)

Tastenfolge: 69 [!]

Ergebnis: 1,711224524E 98 ($1{,}711224524 \times 10^{98}$)

Anmerkung: Bei Berechnungen der Fakultät kann es sehr leicht zur Überschreitung der Rechenkapazität des Computers kommen, was die Fehlermeldung "E" zur Folge hat.

Permutation

Aufgabe: Berechnen Sie die Anzahl der Möglichkeiten, 2 Karten aus 10 Karten, die von 1 bis 10 numeriert sind, auszuwählen und anzuordnen.

Tastenfolge: 10 [nPr] 2 [=]

90.

Kombination

Aufgabe: Berechnen Sie die Anzahl der Möglichkeiten, 2 Karten aus 10 Karten, die von 1 bis 10 numeriert sind, auszuwählen.

Tastenfolge: 10 [nCr] 2 [=]

45.

Dir Prioritätsfolge für verschiedene Berechnungen ist wie folgt:

1. y^x ($\wedge$), nPr, nCr
2. x, ÷
3. +, −

Zufallszahlen

Diese Funktion erzeugt Pseudo-Zufallszahlen zwischen 0,000 und 0,999.

Tastenfolge: [RND]

0.178

[RND]

0.094

Unmittelbar nach Einschalten des Computers werden immer die gleichen Zufallszahlen erzeugt. Wird jedoch unmittelbar nach Einschalten des Computers die Taste [C·CE] gedrückt, werden nicht notwendigerweise die gleichen Zufallszahlen erzeugt.

4.4 EINSATZ ALS BASIC-RECHNER

Rechnen Ohne Programmunterstützung (RUN-mode)

Der Computer kann auch in der Programmiersprache BASIC programmiert werden. Dabei kann er auf zwei verschiedene Arten verwendet werden. Zum einen können Sie Ihre Berechnungen innerhalb eines Programms abarbeiten, zum anderen können die Anweisungen und Funktionen direkt eingegeben und ausgeführt werden (RUN-Mode).

Dabei kann man den Computer wie einen einfachen Taschenrechner benutzen. Um das Ergebnis einer Rechnung zu erhalten muß man anstelle der '='-Taste die [ENTER] -Taste drücken.

Die allgemeine Form einer Rechnung im RUN-Mode ist:

[C-CE] numerischer Ausdruck [ENTER]

Als 'numerischer Ausdruck' wird jede mathematische Formel bezeichnet, die einen Zahlenwert als Ergebnis hat.

Grundrechnungsarten

a) Addition

[5] [Ø] [+] [5] [Ø] [ENTER]

```
100.
```

b) Subtraktion

[1] [Ø] [Ø] [−] [5] [Ø] [ENTER]

```
50.
```

c) Division

[3] [Ø] [Ø] [/] [5] [ENTER]

```
60.
```

d) Multiplikation

[6] [Ø] [*] [1] [Ø] [ENTER]

```
600.
```

e) Klammern

[SHIFT] [(] [6] [7] [5] [+] [6]
[7] [5] [Ø] [SHIFT] [)] [/] [4]
[5] [Ø] [Ø] [Ø] [ENTER]

```
Ø.165
```

Hinweise:

1. Erst nach Drücken der [ENTER] -Taste wird die Rechnung oder das Kommando ausgeführt und das Ergebnis angezeigt.
2. Das Ergebnis der Rechnung kann für weitere Berechnungen verwendet werden. Dazu die Zahl nicht mit der [CA C·CE] -Taste löschen, sondern das nächste Operationszeichen der neuen Formel eingeben (+ − * / ^).

Beispiel:

[3] [0] [0] [*] [1] [5] [0] [ENTER]	45000.
[*] [.] [1] [5]	45000.*.15_
[ENTER]	6750.
[−] [4] [0] [0] [0]	6750.−4000_
[ENTER]	2750.
[−] [1] [2] [2] [5] [ENTER]	1525.
[−] [2] [2] [0] [0] [ENTER]	−675.

Rechnengenauigkeit

Der Computer verarbeitet Zahlen mit einer Genauigkeit bis zu 10 Stellen. Dies führt bei einigen Rechnungen zu Rundungsfehlern. Die Genauigkeitsbeschränkung tritt auch beim Potenzieren oder beim Addieren von Zahlen sehr unterschiedlicher Größe auf.

Beispiele:

Eingabe

```
A = 199 * 199 * 199
```

ENTER

Ausgabe

```
                7880599.
```

Eingabe

```
A – 199 ^ 3
```

ENTER

Ausgabe

```
                Ø.ØØØ24
```

Eingabe

```
1E12+1–1E12
```

ENTER

Ausgabe

```
                Ø.
```

Wissenschaftliche Schreibweise

Man kann Zahlen in den Computer auch im wissenschaftlichen Format eingeben. Damit ist es möglich, Zahlen bis zu einer Größenordung von ±9.999999999E ± 99 einzugeben und zu verarbeiten.

Wird dieser Wertbereich überschritten, erscheint die Fehlermeldung ERROR 2.

Ist der Betrag einer Zahl kleiner als 1 E–99, so wird die Zahl auf Null gesetzt.

Wird eine Zahl mit einem mehr als zweistelligen Exponenten eingegeben, so werden nur die beiden zuletzt eingegebenen Ziffern bewertet.

Hinweis:

1. Wie im englischen Sprachraum üblich, wird nicht das Komma, sondern der Punkt zur Trennung von ganzem und gebrochenem Anteil von Zahlen verwendet.
2. Für den Operator "geteilt durch" wird ein Schrägstrich (/) anstelle des Doppelpunkts gesetzt.
3. Für "multipliziert mit" muß ein Stern (*) gesetzt werden.
4. Als Zeichen für das Potenzieren wird ein Dach (^) verwendet. Ist die Basis negativ, muß sie in Klammern gesetzt werden. Als Exponenten sind bei negativen Grundzahlen nur ganze Zahlen zulässig. Ansonsten erscheint die Fehlermeldung ERROR 2.
5. Sind mehr als 10 Ziffern für die Darstellung eines Ergebnisses erforderlich, wird es im wissenschaftlichen Format angezeigt. Das Anzeigeformat kann durch die BASIC-Anweisung USING verändert werden.

Editier-/Korrekturmöglichkeit

Tippfehler können während der Eingabe oder bei der Überprüfung der Eingaben korrigiert werden.

Mit den Cursor-Tasten [◄] und [►] können Sie den Cursor nach links oder rechts bewegen, ohne daß dabei die eingegebene Formel verändert wird. Zur Korrektur stellen Sie den Cursor über das zu korrigierende Zeichen und überschreiben dieses mit dem neuen Zeichen.

Zum Löschen eines Zeichens den Cursor über das zu löschende Zeichen stellen und die Tasten [SHIFT] und [INS] drücken.

Zum Einfügen von Zeichen drücken Sie die Taste [INS] . Es wird ein Platzhalter (⎵) in die Formel eingefügt, der durch ein neues Zeichen überschrieben werden kann. Durch Drücken der [ENTER] -Taste werden alle überflüssigen Platzhalter aus der Formel gelöscht.

Ist das Ergebnis einer Berechnung schon ausgegeben oder erschien bei der Berechnung eine Fehlermeldung, so kann die Ausgangsformel mit den Tasten [↑] oder [↓] wieder zu Anzeige gebracht und wie oben beschrieben geändert werden.

Hinweis:

BASIC-Schlüsselwörter wie LET, SIN, COS, PRINT usw. werden nach Betätigung der [ENTER] -Taste intern abgekürzt. Obwohl das Schlüsselwort unverändert auf der Anzeige erscheint, wird es beim Editieren durch ein einziges neues Zeichen vollständig überschrieben. Will man sehr lange Programmzeilen eingeben, kommt man vorerst nur bis zum 80. Zeichen. Drücken Sie nun die [ENTER] -Taste und stellen Sie den Cursor an das Zeilenende. Nun können Sie noch zusätzlich einige Zeichen eingeben. Die Eingabe muß wieder mit [ENTER] abgeschlossen werden.

Mathematische Funktionen/Klammerregeln

Der Computer verfügt über alle gebräuchlichen mathematischen Funktionen.

Beispiel:

SHIFT sin Ø ENTER

```
Ø.
```

Die Reihenfolge der Einzelschritte bei der Berechnung eines komplexen mathematischen Ausdrucks wird durch Klammern festgelegt.

Wie in der Mathematik gibt es dabei eine implizite Klammerung, d.h., daß bestimmte Operationen vor anderen den Vorrang haben. Der Computer verfügt über 15 Klammerebenen.

Beispiele:

3 * 5 + 7 * 4

ist gleichbedeutend mit

(3 * 5) + (7 * 4)

Die Rangfolge der mathematischen Operatoren ist:

1. Klammern
2. Abruf von PI, MEM, Variablen
3. Funktionsoperationen in Bezug auf das folgende Argument
4. Potenzen
5. Vorzeichen +, –
6. Multiplikation, Division
7. Addition, Subtraktion
8. Vergleichsoperationen
9. Logische Operationen

Hexadezimal-(Sedezimal-)Zahlen

Natürliche Zahlen im Bereich zwischen 0 und 65535 können auch in sedezimaler Form eingegeben werden.
Der Sedezimalzahl wird dabei ein &-Zeichen vorangestellt. Zur hexadezimalen Darstellung einer Zahl dienen die Ziffern zwischen 0 bis 9 und die Buchstaben A bis F.

Beispiel:

```
&A                                   ENTER
```

Ausgabe

```
                                  10.
```

Eingabe

```
&FFFF                                ENTER
```

Ausgabe

```
                               65535.
```

Textausdrücke

Textausdrücke sind Bestandteil der BASIC-Sprache. Sie können im RUN-Mode ohne Programmunterstützung eingegeben werden.

Man unterscheidet Textkonstanten, Textvariablen, Textfunktionen und zusammengesetzte Texte. Eine Textkonstante ist eine beliebige Zeichenfolge, die durch Anführungszeichen begrenzt ist, wobei die Anführungszeichen nicht Bestandteil der Textkonstante sind.

Beispiel:

C-CE Eingabe

```
"SHARP"                              ENTER
```

Ausgabe

```
SHARP
```

Textvariablen können bis zu sieben Zeichen enthalten. Über die DIM-Anweisung lassen sich Textvariablen bis zu 80 Zeichen erzeugen.

Beispiel:

Eingabe

```
A$="SHARP"
```
ENTER

Ausgabe

```
SHARP
```

Texte können mit Hilfe des "+"-Zeichens aneinandergefügt werden.

Beispiel:

Eingabe

```
A$="POCKET"
```
ENTER

Ausgabe

```
POCKET
```

Eingabe

```
A$+"␣COMPUTER"
```
ENTER

Ausgabe

```
POCKET COMPUTER
```

Hinweis:

Auch das Leerzeichen (Space) innerhalb eines Textes ist ein gültiges Zeichen.

Logische Vergleichsausdrücke

Der Computer kennt folgende Vergleichsausdrücke:

Mathematisches Symbol	BASIC	Deutsche Sprechweise
$<$	<	kleiner als
$\leq$	<=	kleiner gleich
$=$	=	gleich
$\geq$	>=	größer gleich
$>$	>	größer als
$\neq$	<>	ungleich

Mit diesen Operatoren können sowohl zwei numerische als auch zwei Textausdrücke miteinander verglichen werden. Ist die Vergleichsaussage richtig, liefert der Rechner das Ergebnis "1", ist sie falsch, ist das Ergebnis "0". Im Zusammenhang mit der BASIC-Anweisung IF erhalten diese Vergleichsoperatoren eine besondere Bedeutung.

Beispiele:

a) Eingabe

C·CE | 2<1 | ENTER

Ausgabe

```
 0.
```

b) Eingabe

C·CE | 1<=2 | ENTER

Ausgabe

```
 1.
```

c) Eingabe

C·CE | SIN 1 < .7 | ENTER

Ausgabe

0. (im RAD-Mode)
1. (im DEG- und GRAD-Mode)

d) Eingabe

C·CE | "WERNER" > "ANTON" | ENTER

Ausgabe

1.

Die Textausdrücke "WERNER" und "ANTON" werden entsprechend dem ASCII-Code auf die lexikographische Reihenfolge hin überprüft.

Hinweis:

Da das Ergebnis eines logischen Ausdrucks eine Zahl ist, kann dieses als numerische Variable gespeichert und weiterverarbeitet werden.

KAPITEL 5
BASIC - KONZEPT UND BEGRIFFE

5.1 SPRACHELEMENTE

Numerische Konstante

Eine numerische Konstante kann sein:

- eine ganze Zahl (positiv oder negativ)
- eine Dezimalzahl
- eine Zahl in wissenschaftlicher Schreibweise
- eine Sedezimal-(Hexadezimal-)Zahl

Beispiele:

513
−2376
11.745
1.23456788E−12
&AA2B

Textkonstante

Eine Textkonstante ist eine beliebige Zeichenfolge, die durch Anführungszeichen (") begrenzt wird.

Beispiele:

"SHARP"
" " (Textkonstante enthält Leerzeichen)
"" (Textkonstante der Länge 0)

Numerische Variable

Unter einer numerischen Variablen versteht man den Speicherplatz, der zur Aufnahme des jeweiligen, zugehörigen Zahlenwertes bereitsteht.

Der Variablenname setzt sich aus bis zu 2 Zeichen zusammen, wobei das erste Zeichen ein Buchstabe sein muß und das zweite Zeichen eine Ziffer sein kann. Nicht erlaubt sind Sonderzeichen sowie Buchstabenkombinationen, die ein BASIC-Schlüsselwort oder eine Funktion ergeben, z.B. ON, TO, LN, PI etc.
Elemente von Vektoren oder Matrixen werden von ein bzw. zwei in Klammern gesetzte und durch ein Komma getrennte Indexwerte bezeichnet. Indexwerte können auch durch Variablen dargestellt werden.

Beispiele:

A
A(15)
A(1)
A(Z)
AB
A1
XY(2,4)
XZ(A,B)

Textvariable

Unter einer Textvariablen versteht man den Speicherplatz, der zur Aufnahme der jeweiligen, zugehörigen Zeichenfolge bereitsteht.

Der Name der Textvariable ist im Prinzip gleich aufgebaut wie der der numerischen Variable mit einem zusätzlichen "$"-Zeichen hinter dem Buchstaben.

Beispiele:

A$
B$(2)
C$(Z)

Numerische Funktionen, Textfunktionen

Eine numerische Funktion setzt sich aus einem Operator und einem oder mehreren Parametern zusammen. Die Parameter werden hinter den Funktionsnamen gestellt und können je nach Funktion numerische oder Textausdrücke sein. Sind mehrere Parameter erforderlich, so werden diese in Klammern gesetzt und durch Kommata getrennt.

Das Ergebnis einer numerischen Funktion ist ein Zahlenwert, das einer Textfunktion ein Zeichen bzw. eine Zeichenfolge.

Beispiele:

LN60	numerische Funktion; natürlicher Logarithmus des numerischen Parameters 60
ASC"A"	numerische Funktion; Ergebnis ist die numerische Konstante 65, die den ASCII-Code der Textkonstante "A" darstellt.
CHR$65	Textfunktion; Ergebnis ist die Textkonstante "A".
MID$("SHARP",2,2)	Textfunktion; Ergebnis ist die Textkonstante "HA".

Hinweis:

Funktionsoperatoren haben eine höhere Priorität als andere Operatoren.

Beispiel:

SIN A + B = (SIN A) + B

Soll der Sinus der Summe (A + B) gebildet werden, so müssen Klammern gesetzt werden:

SIN (A + B)

Numerischer Ausdrücke

Ein numerischer Ausdruck setzt sich aus einer numerischen Konstanten, Variablen, numerischen Funktion oder deren Verknüpfung durch die arithmetischen Operatoren +, –, *, /, ^, AND, OR und NOT und die Zusammenfassung durch Klammern zusammen.

Beispiele:

15
(–8)
SIN 45
A + B/C
(C * (A * X+B) * 5/(D * C))+10

Textausdrücke

Ein Textausdruck besteht aus einer Textkonstanten, Textvariablen oder einer Textfunktion und deren Verknüpfung durch das "+"-Zeichen. Das Ergebnis eines Textausdrucks ist eine Zeichenfolge.

Beispiele:

"A" + STR$ A
A$ + "PC"

Logische Vergleichsausdrücke

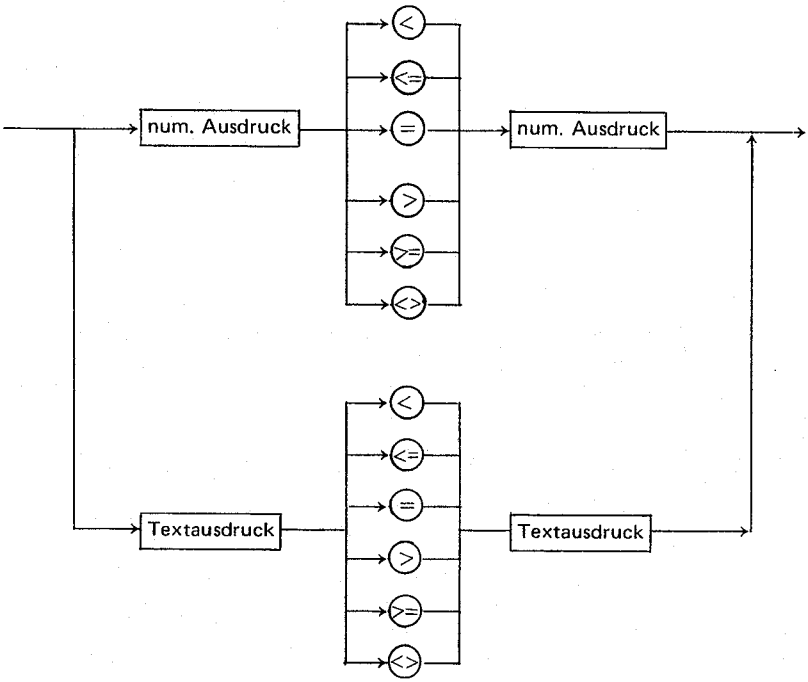
Ein logischer Vergleichsausdruck ist der Vergleich zweier Ausdrücke (numerisch oder Text) durch die Operatoren:

$$>, >=, <, <=, <>$$

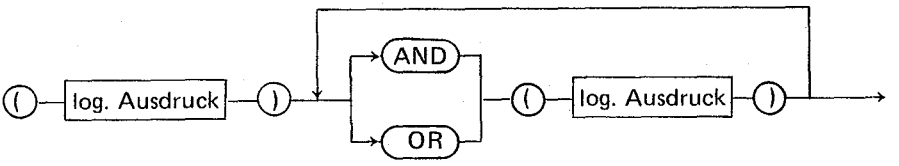
und die Verknüpfung solcher Vergleiche mit den Booleschen Operatoren AND, OR und NOT.

Operation	Mathematisches Symbol	BASIC
Negation	$\neg$	NOT
UND-Funktion	$\wedge$	AND
ODER-Funktion	$\vee$	OR

Syntax:



Logische Vergleichsausdrücke lassen sich wie folgt verknüpfen:



Ist die Vergleichsaussage richtig, so ist das Ergebnis des logischen Ausdrucks "1", ist sie falsch, ist das Ergebnis "0".

Werden Textausdrücke miteinander verglichen, so werden die Textausdrücke auf lexikographische Reihenfolge geprüft; diese richtet sich nach dem ASCII-Code (siehe Anhang).

Werden zwei Textausdrücke mit unterschiedlicher Länge verglichen, so werden beim Vergleich die fehlenden Stellen des kürzeren Ausdrucks mit dem ASCII-Zeichen Null aufgefüllt.

Logische Ausdrücke werden in der BASIC-Anweisung IF verwendet.

Beispiele:

	Ergebnis	
1 < 2	1	
1 + 2 + 3 < 2 + 3 + 4	1	
1 >= 2	0	
1 < 2 < 3	1	
"ANTON" < "WERNER"	1	
"ANTON" > "ANTON 1"	0	("1" ist größer als ASCII-Null)
(1 < 2) AND (3 < 4)	1	
("ANTON" < "WERNER") OR ("ANTON" = "ANTON 1")	1	

Logische Ausdrücke

Logische Ausdrücke sind Relationsausdrücke, die Operatoren AND, OR und NOT. AND und OR verwenden und dienen zur Verbindung von zwei Rationalausdrücken; die Werte der kombinierten Ausdrücke sind in der folgenden Tabelle aufgeführt.

A AND B

		Wert von A	
		Wahr	Falsch
Wert von B	Wahr	Wahr	Falsch
	Falsch	Falsch	Falsch

A OR B

		Wert von A	
		Wahr	Falsch
Wert von B	Wahr	Wahr	Wahr
	Falsch	Wahr	Falsch

Achtung:
Der Wert von A und B muß 0 (falsch) oder 1 (wahr) betragen.
Dezimalzahlen können wie folgt in Binärnotationen von 16 Bit aufgezeichnet werden.

Dezimalzahlen	16-Bit-Binärnotationen
32767	0111111111111111
:	:
3	0000000000000011
2	0000000000000010
1	0000000000000001
0	0000000000000000
−1	1111111111111111
−2	1111111111111110
−3	1111111111111101
:	:
−32768	1000000000000000

Die Negation (NOT) einer Binärnummer 0000000000000001 wird wie folgt vorgenommen:

```
NOT          0000000000000001
(Negation)–  1111111111111110
```

1 wird zu 0 invertiert und 0 zu 1. Dies nennt sich "Notation (NOT-Betrieb)". Wenn 1 und NOT 1 addiert werden, ergibt sich dann folgendes Resultat:

```
  0000000000000001 (1)
+) 1111111111111110 (NOT 1)      Einerkomplement
  1111111111111111 (–1)          Zweierkomplement
```

Alle Bits werden deshalb 1. Gemäß der obigen Zahlenliste werden die Bits – 1 in Dezimalnotationen, d.h.,

$$1 + \text{NOT } 1 = -1$$

Das Verhältnis zwischen dem numerischen Wert X und dessen negiertem (oder invertiertem) Wert NOT X ist:

$$X + \text{NOT } X = -1$$

Dieses Ergebnis ist eine Gleichung von NOT X = –X–1, bzw.

$$\text{NOT } X = -(X + 1)$$

Aus dieser Gleichung ergeben sich folgende Resultate:

$$\text{NOT } 0 = -1$$
$$\text{NOT } -1 = 0$$
$$\text{NOT } -2 = 1$$

Mit diesen Operatoren können mehr als zwei relationale Ausdrücke kombiniert werden. Um den beabsichtigten Vergleich deutlich zu machen, empfiehlt sich die Verwendung von Klammern.

```
(A<9) AND (B>5)
(A≥10) AND NOT (A>20)
(C=5) OR (C=46) OR (C=7)
```

Der Computer implementiert logische Operatoren als "bitweise" logische Funktionen an 16-Bit-Quantitäten. (Siehe Anmerkung bezüglich relationaler Ausdrücke und Wahr und Falsch.) Bei normalem Betrieb ist dies unwesentlich, da die einfachen 1 und 0 (Wahr und Falsch), die aus einem relationalen Ausdruck resultieren, nur mit einem Bit arbeiten. Bei Anwendung eines booleschen Operators auf einen anderen Wert als 0 oder 1 funktioniert dies unabhängig an jedem Bit. Wenn zum Beispiel A 17 und B 22 ist, dann ist (A OR B) 23:

```
17 OR 22 ist 10001 .. 17
             10110 .. 22   OR Betrieb
             10111 .. 23 in Dezimalzahl
```

17 und 22 werden zuerst in Binärzahlen umgewandelt. Für jede Ziffer, wird, wenn jedes Bit 1 ist, der boolesche Operator 1 gelassen Andernfalls wird der boolescher Operator 0 gelasseu.

Für einen geübten Programmierer wird sich dieser Betriebsablauf gelegentlich als sehr nützlich erweisen. Anfänger sollten sich an klare und einfache Wahr-oder Falsch-Vergleichsausdrücken halten.

5.2 VARIABLE

Variablen sind Bezeichner für Größen, deren Were erst im Laufe des Programmablaufs festgelegt werden. Sie werden beim Rechner durch Speicherplätze realisiert, denen man unterschiedliche Daten durch Einlesen oder durch Auswerten eines bestimmten Ausdrucks zuordnen kann. Je nachdem, ob es sich bei der Variablen um einen numerischen oder um einen Textausdruck handelt, nennt man solche Variablen numerische Variablen oder Textvariablen.

Zur Unterscheidung zwischen numerischen und Textvariablen muß jeder Name einer Textvariablen mit dem "$"-Zeichen enden. Beide Typen können als einfache oder indizierte Variable geschrieben werden. Zur leichteren Identifizierung erhalten sie einen Variablennamen als symbolische Adresse für den Speicherplatz. Mit Variablen lassen sich Gesetzmäßigkeiten unabhängig von ihrem jeweiligen Wert allgemein ausdrücken.

Im Computer stehen folgende zwei verschiedene Speicherbereiche für Variablen zur Verfügung.

a) Standardvariablenspeicher
b) Feldvariablenspeicher

Der Standardvariablenspeicher ist ein begrenzter Speicherbereich und dient ausschließlich zur Aufnahme der 26 Standardvariablen.

Der Feldvariablenspeicher ist Bestandteil des Hauptspeichers, der auch zur Aufnahme von Programmen dient. Die Kapazität des Hauptspeichers kann wahlweise für Programme und für Feldvariablen benutzt werden.

Folgende Variablentypen stehen zur Verfügung:

1. Standardvariablen
   a) einfache numerische Variable
   b) einfache Textvariable
   c) indizierte numerische Variable
   d) indizierte Textvariable

2. Feldvariablen
   a) indizierte eindimensionale numerische Feldvariable (Vektor)
   b) indizierte eindimensionale Textfeldvariable (Vektor)
   c) indizierte zweidimensionale numerische Feldvariable (Matrizen)
   d) indizierte zweidimensionale Textfeldvariable (Matrizen)

Standardvariable

Für die Standardvariablen stehen 26 reservierte Speicherbereiche zur Verfügung, die wahlweise als numerische oder als Textvariable belegt werden können.

Den Standardvariablen können also wahlweise numerische oder Textwerte zugeordnet werden. Bei der Wertabfrage der Standardvariablen muß der Variablenname dem Inhalt dieser Variablen entsprechen. Wird z.B. eine numerische Variable als Textvariable aufgerufen, und umgekehrt, so wird am Display die Fehlermeldung ERROR 9 angezeigt.

Standardvariable:

A = A$ = A(1) = A$(1)
B = B$ = A(2) = A$(2)
C = C$ = A(3) = A$(3)
D = D$ = A(4) = A$(4)
E = E$ = A(5) = A$(5)
F = F$ = A(6) = A$(6)
G = G$ = A(7) = A$(7)
H = H$ = A(8) = A$(8)
I = I$ = A(9) = A$(9)
J = J$ = A(10) = A$(10)
K = K$ = A(11) = A$(11)
L = L$ = A(12) = A$(12)
M = M$ = A(13) = A$(13)
N = N$ = A(14) = A$(14)
O = O$ = A(15) = A$(15)
P = P$ = A(16) = A$(16)
Q = Q$ = A(17) = A$(17)
R = R$ = A(18) = A$(18)
S = S$ = A(19) = A$(19)
T = T$ = A(20) = A$(20)
U = U$ = A(21) = A$(21)
V = V$ = A(22) = A$(22)
W = W$ = A(23) = A$(23)
X = X$ = A(24) = A$(24)
Y = Y$ = A(25) = A$(25)
Z = Z$ = A(26) = A$(26)

Einfache Variablen

Einfache Variablen werden mit den Namen A bis Z als numerische Variable bzw. A$ bis Z$ als Textvariable aufgerufen.

Einer einfachen numerischen Variablen kann man eine Zahl mit maximal 10 Stellen, einen zweistelligen Exponenten und die Vorzeichen zuordnen.

Beispiele:

```
A = 123
A = SIN X
A = B + C
A = B * C
```

Einer Textvariablen kann man Zeichenfolgen bis zu sieben Zeichen, bestehend aus Buchstaben, Zahlen, Sonderzeichen und Leerstellen, zuweisen. Bei der Wertzuweisung muß der Text durch Anführungszeichen begrenzt werden.

Beispiele:

```
B$="TEXT"
B$="T  E  X  T"
```

Indizierte Variable

Indizierte Variablen werden mit den Namen A(1) bis A(26) als numerische bzw. mit A$(1) bis A$(26) als Textvariable aufgerufen.

Die indizierten Standardvariablen sind äquivalent zu den oben beschriebenen einfachen Standardvariablen. So entspricht z.B. die einfache Variable A der indizierten Variablen A(1), die einfache Variable B der indizierten Variablen A(2); usw.

Die indirekte Adressierung (Indizierung) einer Variablen gestattet es, den Namen und die Adresse einer Variablen in Abhängigkeit vom Wert einer anderen Variablen, der z.B. ein Rechenergebnis sein kann, festzulegen. Außerdem kann der Index einer Variablen auch das Ergebnis eines numerischen Ausdrucks sein.

Beispiele:

```
LET A(A) = 1243
```

Hiermit wird eine numerische Variable definiert, deren Index gleich dem ganzzahligen Anteil der im Speicher A stehenden Zahl ist. Angenommen, der Wert der Variablen A ist 7, so wird der Variablen A(7), also der Variablen G, der Wert 1234 zugewiesen.

LET A(B/5) = 789

Hierbei wird eine numerische Variable definiert, deren Index gleich dem ganzzahligen Anteil des Quotienten aus B/5 ist.

Angenommen, der Wert der Variablen B ist 134, so ist der Quotient aus 134/5 = 26,8, und somit wird der Variablen A(26) = Z ein Wert von 789 zugeordnet.

LET A$(A) = "TEXT"

Der Wert der numerischen Variablen A definiert den Index der Textvariablen A$. Dabei muß der Wert der Variablen größer als 1 sein, da die Variable A$(0) nicht existiert und die Variable A$(1) der Variablen A$ entspricht und die beiden, wie bereits erklärt, nicht gleichzeitig definiert werden können.

Besonderheiten der Variablen A

Die Variable A kann als:

a) Standardvariable A bzw. A$
b) Indizierte Standardvariable A(n) bzw. A$(n)
   (n = 1 bis 26)
c) Eindimensionale Feldvariable A(n) bzw. A$(n)
   (n = 0 bis 255)
d) Zweidimensionale Feldvariable A(n, n) bzw. A$(n, n)
   (n = 0 bis 255)

definiert werden.

Dabei gelten folgende Grundsätze:

1) Die Variable A ohne Dimensionierung

1. Diesen A-Variablen können nur numerische Werte mit zehnstelliger Mantisse, zweistelligem Exponenten und einem Vorzeichen bzw. Textwerte mit max. 7 Zeichen zugewiesen werden. Sie können nur alternativ als numerische oder Textvariablen definiert werden.
2. Der Variablenname A(0) ist unzulässig.
3. Für die indizierte Variable A(n), (n = 1 bis 26) = Standardvariablen A bis Z, wird kein Speicherplatz im Hauptspeicher reserviert.
4. Für die indizierte Variable A(n), (n = 27 bis 255) = eindimensionaler Feldvariable, wird ein Speicherplatz im Hauptspeicher reserviert. Der Platz wird automatisch durch den höchsten definierten Indexwert reserviert. Wird z.B. A(n), n = 50, aufgerufen, ist der A-Vektor mit den Elementen A(27) bis A(50) automatisch vereinbart.

5. Der A-Vektor wird bezüglich seiner Elementeanzahl automatisch auf den Wert begrenzt, den er vor der Dimensionierung eines weiteren Vektors bzw. einer Matrix hatte.

6. Bei der ersten Verwendung von Indizes größer als A(27) oder A$(27) werden 7 Bytes für den Variablennamen und 8 Bytes für jede Variable verwendet.

Beispiel:

```
 5: A(27) = 27
10: DIM B (10)
15: DIM A (40)      oder 15: A(40) = 40
```

Nach dem Starten des Programms wird die Fehlermeldung ERROR 3 angezeigt. Der A-Vektor wurde durch die Dimensionierung des B-Vektors auf ein Element, nämlich A(27), begrenzt.
Durch den Anruf von A(40) in Zeile 15 müßte ein zweiter A-Vektor definiert werden, was aber unzulässig ist.

7. Die Variablen A(n) bzw. A$(n) (n = 1 bis 26) können nicht gelöscht werden. Mit den Kommandos NEW bzw. CLEAR wird ihr Wert auf 0 (Null) bzw. auf das ASCII-Zeichen Null gesetzt.
   Variablen A(n) bzw. A$(n) werden durch das Kommando RUN gelöscht.

2) Die Variable A mit Dimensionierung

1. Nach einer Dimensionierung der Variablen A kann auf die Standardvariablen nur noch direkt zugegriffen werden. Nach z.B. DIM A(26) entspricht A(1) nicht mehr der Standardvariablen A und A(26) nicht mehr der Standardvariablen Z.
   Das Element A(1) und die Standardvariable A haben nun verschiedene Speicherplätze.

2. Wird nach dem Aufruf einer undimensionierten Standardvariablen, z.B. A(2) = B eine Dimensionierung, z.B. DIM A(30) durchgeführt, wird ERROR 3 angezeigt.

Feldvariable

Feldvariablen können als eindimensionale Felder (Vektoren) oder als zweidimensionale Felder (Matrizen) definiert werden, die wiederum unterschiedlich viele Elemente haben können.
Die Feldvariablen werden im Hauptspeicher abgelegt und müssen vor ihrem Aufruf wegen ihrer flexiblen Länge dimensioniert werden. Weitere Einzelheiten siehe unter dem BASIC-Befehl DIM (Kapitel 10).

Der Speicherplatzbedarf für eine Feldvariable setzt sich zusammen aus:

7 BYTES für den Variablennamen
8 BYTES für ein numerisches Element
16 BYTES für ein Textelement im Standardformat

Löschen von Variablen

(1) Löschen fester und einfacher Variablen
Beispiel: CLEAR A, AD, D1$
Die feste Variable (A) wird 0 oder Null, und die einfachen Variablen (AD, D1$) werden gelöscht.
Der freie Bereich wird entsprechend der gelöschten Variablen vergrößert.

(2) Löschen von Feldvariablen
Beispiel: ERASE X, Z$
Mit einer oder zwei Dimensionen spezifizierte Feldvariable (X( ), Z$( )) werden gelöscht. Auch der Bereich für feste Variablen mit Indizes über 26 wird gelöscht.
(Siehe den Abschnitt "Besonderheiten der Variablen A".)

(3) Löschen aller Variablen
Beispiel: CLEAR
Durch diese Ausführung werden die einfachen Variablen und die mit dem DIM-Befehl deklarierten Feldvariablen sowie alle Inhalte der festen Variablen gelöscht.

Hinweis: Programme und Kalkulationstabellen-Daten werden mit dem CLEAR-Befehl nicht gelöscht.

Numerische Feldvariable

Numerische Feldvariablen können sowohl als Vektoren als auch als Matrizen definiert werden. Jedem Element der Vektoren bzw. der Matrizen kann eine vollständige Zahl, bestehend aus einer zehnstelligen Mantisse, einem zweistelligen Exponenten und einem Vorzeichen zugeordnet werden.

Vektoren werden mit dem Vektorennamen und dem Index für ein Element aufgerufen.

Der Vektorenname kann aus einem oder aus zwei Zeichen bestehen. Das erste Zeichen muß ein Buchstabe sein, das zweite Zeichen kann ein Buchstabe oder eine Ziffer sein.

Buchstabenkombinationen, die ein BASIC-Schlüsselwort oder eine Funktion ergeben, sind als Variablenname nicht erlaubt, z.B. LN, ON, OR, PI etc.

Beispiel:

B(0)
C1(3)
DF(80)

Matrizen werden mit dem Matrizennamen und den beiden Indizes für ein Element aufgerufen.

Der Matrizenname kann aus einem oder aus zwei Zeichen bestehen. Das erste Zeichen muß ein Buchstabe sein, das zweite Zeichen kann ein Buchstabe oder eine Ziffer sein.

Buchstabenkombinationen, die ein BASIC-Schlüsselwort oder eine Funktion ergeben, sind als Variablenname nicht erlaubt, z.B. LN, ON, OR, PI etc.

Beispiel:

D(2,4)
E1(1,1)
YZ(0,0)

Insgesamt stehen für Vektoren und Matrizen folgende Namen zur Verfügung:

Alle Einzelbuchstaben oder alle Zweierkombinationen aus den Buchstaben A – Z und den Ziffern 0 – 9, wobei das erste Zeichen ein Buchstabe sein muß.

Die Indizes für die Elemente liegen theoretisch im Intervall 0 – 255.
Tatsächlich ist das Intervall jedoch abhängig von der aktuell freien Speicherkapazität des Hauptspeichers.

Buchstabenkombinationen, die ein BASIC-Schlüsselwort oder eine Funktion ergeben, sind als Variablenname nicht erlaubt, z.B. LN, ON, OR, PI etc.

Der Index für das Vektoren- und Matrizenelement kann als numerische Konstante oder durch eine Variable definiert werden.

Beispiele:

B(6) bzw. B(A)
H1(3,5) bzw. H1(E,B)

Der Index wird durch den jeweiligen Wert der Variablen ausgedrückt.

Vektoren und Matrizen dürfen nicht mit dem gleichen Feldnamen und nicht zweimal definiert werden. Die Namen B(n) und B(n,n) sind also nicht gleichzeitig zulässig.

Hinweis:

Bei Verwendung der Feldvariablen 'A' beachten Sie den Abschnitt "Besonderheiten der Variablen A".

Textfeldvariable

Textfeldvariablen können wie die numerischen Feldvariablen als Vektor oder als Matrix definiert werden. Jedem Element der Vektoren oder Matrizen können im Standardformat Texte mit maximal 16 Zeichen zugeordnet werden. Durch eine entsprechende Dimensionierung (siehe DIM-Anweisung in Kapitel 10) kann die Größe der Elemente zwischen 1 und 80 Zeichen frei festgelegt werden.

Beispiel: DIM B$(10) * 4

In diesem Beispiel können für die Elemente des B$-Vektors (B$(0) bis B$(10)) jeweils nur 4 Zeichen eingegeben werden.

Insgesamt stehen für die Vektoren und Matrizen folgende Feldnamen zur Verfügung:

Alle Einzelbuchstaben oder alle Zweierkombinationen aus den Buchstaben A — Z und aus den Ziffern 0 — 9, wobei das erste Zeichen ein Buchstabe sein muß.
Buchstabenkombinationen, die ein BASIC-Schlüsselwort oder eine Funktion ergeben, sind als Variablenname nicht erlaubt, z.B. LN, ON, OR, PI etc.
Die Indizes für die Elemente liegen theoretisch im Intervall 0 — 255.
Tatsächlich ist das Intervall jedoch abhängig von der aktuell freien Speicherkapazität des Hauptspeichers.

Vektoren werden mit dem Vektorennamen und dem Index für ein Element aufgerufen.

Beispiele:

B$(0)
A5$(0)
Z$(80)

Matrizen werden mit dem Matrizennamen und den beiden Indizes für ein Element aufgerufen.

Beispiele:

D$(2,4)
Z1$(5,10)

Die Indizes für das Vektoren- und Matrizenelement können als numerische Konstante oder als numerische Variable definiert sein, wobei der Index durch den Wert der jeweiligen Variablen bezeichnet wird.

Beispiele:

C$(A)
AE$(E,F)

Vektoren und Matrizen dürfen nicht mit dem gleichen Feldnamen und nicht zweimal definiert werden. Die Namen B$(n) und B$(n,n) sind also nicht gleichzeitig zulässig.

5.3 ARITHMETISCHE FUNKTIONEN

Der Computer bietet eine große Anzahl von arithmetischen Standard-Funktionen. Hierzu gehören:

Trigonometrische und ihre Umkehrfunktionen
Natürlicher und dekadischer Logarithmus und Umkehrfunktionen
Polarkoordinatenberechnungen

Die Ergebnisse der Funktionen sind numerische Werte. Als Parameter ist im allgemeinen ein numerischer Ausdruck zugelassen.

Die meisten Funktionsnamen können über die Funktionstasten des Rechnerteils eingegeben werden:

Funktionsname		Funktion
ABS		Absolutwert
ACS	SHIFT cos⁻¹	Arcuscosinus
ASN	SHIFT sin⁻¹	Arcussinus
ATN	SHIFT tan⁻¹	Arcustangens
COS	SHIFT cos	Cosinus
CUR	C U R	Kubikwurzel
EXP	SHIFT e^x	Exponentialumrechnung
FACT	F A C T	Fakultät
INT		Ganzzahlfunktion
LN	SHIFT ln	Natürlicher Logarithmus
LOG	SHIFT log	Dekadischer Logarithmus
MEM		Freier Speicherplatz
NCR	SHIFT nCr	Kombination
NPR	SHIFT nPr	Permutation
PI	P I	Kreiskonstante PI
RADIAN		Winkeleinheit
RANDOM		Zufallsgeneratoranfangswert
RCP	R C P	Reziprokwert
RND		Pseudo-Zufallszahl
ROT	SHIFT $\sqrt[x]{y}$	Wurzelberechnungen (x-te Wurzel)
SGN		Vorzeichenfunktion
SIN	SHIFT sin	Sinus
SQU	x^2	Quadratzahl
SQR	$\sqrt{\ }$	Quadratwurzel
TAN	SHIFT tan	Tangens
TEN	SHIFT 10^x	Exponentialfunktion 10^x
^	SHIFT $y^{(x)}$	Potenzfunktion

5.4 TEXTFUNKTIONEN

Mit dem Computer können Sie nicht nur rein numerische Aufgaben lösen, sondern auch Texte verarbeiten. Die im folgenden Kapitel beschriebenen Funktionen sollen Ihnen diese Arbeit erleichtern. So können Sie beispielsweise Zeichenfolgen aus Texten heraustrennen und zu neuen zusammensetzen.

Die Ergebnisse der Textfunktionen sind entweder Zeichenketten, die Textvariablen zugewiesen oder in Textausdrücken weiterverarbeitet werden können, oder numerische Werte, die sich in numerischen Ausdrücken verarbeiten oder numerischen Variablen zuordnen lassen.

Folgende Textfunktionen stehen zur Verfügung:

ASC	Ermittelt von einem ASCII-Zeichen den zugehörigen ASCII-Code.
CHR$	Ermittelt von einem ASCII-Code das zugehörige ASCII-Zeichen.
LEFT$	Entnimmt einer Zeichenfolge eine spezifizierte Zeichenanzahl von links.
RIGHT$	Entnimmt einer Zeichenfolge eine spezifizierte Zeichenanzahl von rechts.
MID$	Entnimmt einer Zeichenfolge eine spezifizierte Zeichenanzahl aus der Mitte.
STR$	Wandelt den Wert eines numerischen Ausdrucks in eine Zeichenfolge um.
VAL	Wandelt eine als Zeichenfolge eingegebene Zahl in ihren numerischen Wert um.
LEN	Berechnet die Anzahl der Zeichen eines Textausdruckes.

KAPITEL 6
PROGRAMMIEREN IN BASIC

Der Computer verwendet die weit verbreitete Programmiersprache BASIC (Beginner's All-Purpose Symbolic Instruction Code). Diese Programmiersprache wurde Anfang der 60er Jahre am Dartmouth College entwickelt und hat sich bis heute insbesondere bei den Mikrocomputern durchgesetzt.

BASIC ist im Gegensatz zu anderen Programmiersprachen einfach und leicht zu erlernen.

BASIC gestattet den Dialogbetrieb mit dem Rechner und ist dabei so flexibel, daß ein bestehendes Programm ohne großen Aufwand geändert werden kann.

Abgesehen von einigen Abweichungen von BASIC-Version zu BASIC-Version ist diese Programmiersprache maschinenunabhängig.

6.1 BASIC – ÜBERSICHT

Wie jede natürliche Sprache hat auch die formale Computersprache BASIC eine Grammatik und einen zugehörigen Zeichensatz.
Letzterer setzt sich aus folgenden Zeichen zusammen:

Buchstaben: A, B, ... , Z
Zahlen: 0, 1, ... , 9
Sonderzeichen: ( ) . ; , : + – * / $ & % # @ ! ? < > = ^

Zur Vermeidung von Verwechslungen unterscheidet man zwischen Ø (Null) und dem Buchstaben O.

BASIC kennt folgende Sprachelemente:

- numerische Konstanten
- Textkonstanten
- numerische Variablen
- Textvariablen
- dimensionierte oder Feldvariablen
- BASIC-Schlüsselwörter für
  - Standardfunktionen
  - Zuweisungen
  - Eingaben
  - Ausgaben
  - Steuerungen
  - Kommandos

– Operationszeichen (+, –, *, /, ^, >, <, =, >=, <=, <>)

Diese Sprachelemente werden zu Programmsätzen zusammengefügt, die vom Rechner in der angegebenen Reihenfolge abgearbeitet werden.

Programmerstellung

Die Programmerstellung gliedert sich im wesentlichen in drei Schritte:

PROBLEMANALYSE

UMSETZUNG IN DIE PROGRAMMIERSPRACHE
UND EINGABE IN DEN COMPUTER

TESTEN DES PROGRAMMS

Problemanalyse:

Das gestellte Problem muß zunächst analysiert und dann so aufbereitet werden, daß sich die einzelnen Teilprobleme leicht und übersichtlich programmieren lassen.

Umsetzung in die Programmiersprache und Eingabe in den Computer:

Die in die einzelnen Schritte zerlegte Aufgabenstellung wird in eine Programmiersprache (hier BASIC) übertragen und in den Computer eingegeben.

Testen des Programms:

Nur sehr selten wird das umgesetzte und eingegebene Programm auf Anhieb fehlerfrei arbeiten. Dabei können die verschiedenartigsten Fehler auftreten:

– Schreibfehler
– falsche Sprachelemente (SYNTAX-Fehler)
– falscher logischer Aufbau
– falsche Programmstruktur (fehlerhafte Problemanalyse)

Der Computer bietet einige Möglichkeiten, auftretende Fehler zu finden und zu beheben (siehe Abschnitt Fehlermeldungen/Fehlersuche).

Programmaufbau

Bei der Übertragung der Problemanalyse in ein BASIC-Programm sind die im folgenden aufgeführten Regeln zu beachten:

- Ein vollständiges BASIC-Programm besteht aus einer Reihe von Anweisungen, die in der Reihenfolge angeordnet sein müssen, in der sie später ausgeführt werden sollen. Eine Ausnahme hierfür sind die GOTO-, GOSUB-, ON GOTO- und ON GOSUB-Anweisungen, die das Programm an eine festgelegte Stelle verzweigen können.

- Die Programmierung erfolgt zeilenweise, wobei jede Programmzeile eine oder mehrere durch Doppelpunkt getrennte Programmanweisungen enthalten kann.

- Die Anweisungen dürfen nicht länger als eine Zeile sein, d.h., sie können nicht in der nächsten Zeile fortgesetzt werden. Eine Zeile kann bis zu 79 Zeichen enthalten, wovon jeweils 16 Zeichen auf dem Display angezeigt werden. Die restlichen Zeichen einer Zeile können mit den (Cursor-)Tasten [►] oder [◄] sichtbar gemacht werden.

  Hinweis: BASIC-Schlüsselwörter wie LET, SIN, COS, PRINT usw. werden nach Betätigung der [ENTER] -Taste intern abgekürzt. Bei der Eingabe einer sehr langen Programmzeile kann man vorerst nur 79 Zeichen eingeben. Nach Drücken der [ENTER] -Taste werden die Schlüsselwörter abgekürzt, und man kann zusätzlich noch einige Zeichen eingeben. Die Eingabe muß wieder mit [ENTER] abgeschlossen werden.

- Jede Zeile muß mit einer positiven ganzen Zahl (der Zeilennummer) beginnen. Eine Zeilennummer darf nur einmal im Programm verwendet werden. Die Programmzeilen werden vom Rechner in aufsteigender Reihenfolge ausgeführt, wenn nicht durch Steueranweisungen (z.B. GOTO) eine andere Reihenfolge festgelegt wird.
  Die Numerierung muß nicht lückenlos sein, es ist sogar zweckmäßig, z.B. Zehnerschritte zu wählen, da damit die Möglichkeit geboten ist, nachträglich noch Programmzeilen einzufügen.
  Die Zeilennummern dürfen im Bereich von 1 bis 65279 gewählt werden.

Eingabe eines Programms

Die Programmeingabe erfolgt im PRO-Mode. Ein eventuell noch im Speicher befindliches Programm kann mit der NEW-Anweisung gelöscht werden.
Mit dieser Anweisung werden jedoch nicht die Daten der Kalkulationstabellen gelöscht.

Beispiel einer Programmeingabe:

```
NEW
10 A = 15
20 B = 3
30 C = A + B
40 PRINT C
50 END
```

Jede Programmzeile muß mit [ENTER] abgeschlossen werden. Der Rechner fügt dann einen Doppelpunkt zwischen die Zeilennummer und der ersten Anweisung ein.

Beispiel:

10 A = 15 [ENTER]

Anzeige:

10: A = 15

Überprüfen des Programms, Editieren und Auflisten

Ist die Programmeingabe abgeschlossen, so kann man im PRO-Mode den Inhalt der einzelnen Zeilen noch einmal überprüfen.

Betätigt man die [↑] -Taste, wird die vorhergehende Programmzeile angezeigt.
Betätigt man die [↓] -Taste, wird die folgende Programmzeile angezeigt.

Mit der LIST-Anweisung kann man genau spezifizierte Zeilen zur Anzeige bringen.

Beispiele:

LIST	Die erste Zeile eines Programms wird angezeigt.
LIST 100	Die Programmzeile 100 wird angezeigt.

Mit der LLIST-Anweisung kann das Programm über den Drucker (Option CE-126P) auf Papier 'gelistet' werden.

Beispiele:

LLIST	Das gesamte Programm wird ausgedruckt.
LLIST 100	Die Programmzeile 100 wird ausgedruckt.
LLIST 100,200	Die Programmzeilen von 100 bis 200 werden ausgedruckt.
LLIST 100,	Das Programm wird ab Programmzeile 100 ausgedruckt.
LLIST ,100	Das Programm wird bis Programmzeile 100 ausgedruckt.

Beispiel:

```
10 A = 15
20 B = 3
30 C = A + B
40 PRINT C
50 END
```

Durch Eingabe von LIST 30 (nur im PRO-Mode möglich) wird am Display folgendes angezeigt:

30: C = A + B

Die Zeile 40 kann jetzt durch Drücken der [↓] -Taste angezeigt werden.

40: PRINT C

Korrektur einer Zeile

Die Korrektur einer Zeile (nur im PRO-Mode) ist sehr einfach und kann auf zwei verschiedene Arten erfolgen:

1. Überschreiben der Zeile:
   Man gibt die neue Zeile mit derselben Zeilennummer ein. Die alte Version der Programmzeile wird dadurch überschrieben.

2. Editieren:
   Man bringt die zu editierende Zeile mit LIST zur Anzeige und drückt eine der beiden Cursor-Tasten ([◄] oder [►]). Damit wird der Doppelpunkt zwischen der Zeilennummer und der ersten Programmanweisung unterdrückt und die Zeile zur Korrektur freigegeben. Jetzt können Sie mit Hilfe der [SHIFT] [DEL] - oder der [INS] -Taste Zeichen in der Zeile löschen oder einfügen. Auch ein Überschreiben einzelner Zeichen ist möglich.

Beispiel:

In dem oben angeführten Beispiel soll die Zahl 15 durch die Zahl 17 ersetzt werden. Hierzu geben Sie ein:

LIST 10 [ENTER] (Zeile 10 anzeigen)

[►] (Beginn Änderung)

[►] [►] [►] (Cursor auf die '5')

7 [ENTER] ('5' durch '7' ersetzen)

Löschen und Kopieren einer Zeile

Löschen einer Zeile

Das Löschen einer Programmzeile erfolgt im PRO-Mode. Ein Zeile wird ersatzlos gelöscht, wenn man nur die Zeilennummer eingibt und die [ENTER] -Taste drückt.

Beispiel:

```
10  A = 15
20  B = 3
30  C = A + B
40  PRINT C
50  END
```

Eingabe:

```
10 [ENTER]
20 [ENTER]
```

Listet man nun das Programm, so sieht man, daß die Zeilen 10 und 20 gelöscht wurden.

Kopieren einer Zeile

Beim Programmieren kommt es häufig vor, daß mehrere Programmzeilen fast den gleichen Inhalt haben oder daß Programmzeilen im ersten Programmteil auch in einem weiteren Programmteil benutzt werden können. Um sich die Arbeit der Eingabe zu vereinfachen, kann der Inahlt einer bestehenden Programmzeile auf eine oder weitere neue Zeilennummern kopiert werden. Die alte Zeilennummer und ihr Programm bleibt unverändert.

Das Kopieren einer Programmzeile erfolgt im PRO-Modus. Man bringt die zu kopierende Zeile mit "LIST" oder den Tasten [↑] , [↓] zur Anzeige. Mit den Cursor-Tasten wird der Cursor auf die Zeilennummer gebracht. Nun kann die Zeilennummer überschrieben werden, gegebenenfalls unter Zuhilfenahme der Tasten SHIFT [←] oder SHIFT [→] . Nach [ENTER] wird die kopierte Zeile mit der neuen Zeilennummer angezeigt und kann ggf. editiert werden.

Programmausführung

Die Programmausführung kann nur im RUN-Mode erfolgen. Der Programmstart kann durch drei verschiedene Anweisungen erfolgen (siehe RUN/GOTO/Definable Keys).

Nach Eingabe des Startkommandos beginnt der Computer mit der Abarbeitung der einzelnen Programmzeilen. Während der Programmausführung ist auf der Anzeige das Wort "BUSY" sichtbar. Dies erlischt entweder am Programmende oder wenn eine Anzeige am Display erfolgt. Solange 'BUSY' aufleuchtet, nimmt der Computer keine Eingaben von der Tastatur an. Nach Beendigung des Programms wird am Display das 'Bereitschaftssymbol' (>) angezeigt.

Zur Unterbrechung einer Programmausführung dient die [BRK] -Taste. Am Display wird folgendes angezeigt:

BREAK IN XX wobei XX, die Zeilennummer ist, die vor der Unterbrechung bearbeitet wurde.

Will man nach einer Unterbrechung das Programm fortsetzen, so genügt es, die CONT(inue)-Anweisung einzugeben:

CONT

Die Programmausführung wird fortgesetzt.

Fehlermeldung/Fehlersuche

Tritt während der Programmausführung ein Fehler auf, so wird dieser erkannt und gemeldet. Die Fehlermeldung wird am Display in folgender Form angezeigt:

ERROR Fehlercode IN Zeilennummer

Die Liste der Fehlermeldungen und ihre Erklärung finden Sie numerisch geordnet im Anhang.

Gleichzeitig mit dem Fehlercode wird die Nummer der Zeile, in der der Fehler auftrat, angezeigt. Diese Zeile kann nun auch im RUN-Mode am Display zur Anzeige gebracht werden. Löschen Sie die Fehlermeldung mit der [C·CE] -Taste und drücken Sie die Taste [↑] . Die Zeile erscheint auf der Anzeige, und der Cursor blinkt auf dem fehlerhaften Sprachelement. Will man die Zeile korrigieren, muß man in den PRO-Mode umschalten und wie im Abschnitt "Korrektur einer Zeile" beschrieben vorgehen.

Beispiel:

```
10 A = 2
20 B = 10
30 C = B/A
40 PRINT C,
50 END
```

Bei der Ausführung diese Programm erscheint:

ERROR 1 IN 40

Die PRINT-Liste in Zeile 40 ist unvollständig. Schreiben Sie hinter das Komma ein A.

40 PRINT C,A

Das Programm ist jetzt fehlerfrei.

KAPITEL 7
PRAKTISCHE HINWEISE

Der Computer verfügt über einige Erleichterungen zur Eingabe von Programmen. Eine dieser Erleichterungen ist die Verwendung von Abkürzungen für Befehle und Kommandos (siehe Kapitel 10).
Dieses Kapitel beschreibt ein Merkmal, des unnötige Eingabe vermeidet — die Taste [DEF]

7.1 DIE DEF-TASTE UND ETIKETTIERTE PROGRAMME

Oft sind gleichzeitig mehrere Programme im Speicher des Computers vorhanden (die über unterschiedliche Zeilennummern verfügen müssen). Um normalerweise ein Programm mit dem Kommando RUN oder GOTO zu starten, muß man die Anfangszeilennummer jedes Programms wissen (siehe Kapitel 10). Aber es gibt auch eine einfachere Möglichkeit.
Jedes Programm kann mit einem Etikett versehen und dann durch nur zwei Tastenbetätigungen ausgeführt werden.
Dafür wird die DEF-Taste verwendet.

Hinweis: Die erste Zeile des Programms erhält ein Etikett, das aus einem einzelnen, in Anführungsstriche gestellten Buchstaben besteht, auf den ein Doppelpunkt (:) folgt.

```
10: "A": PRINT "FIRST"
20: END
30: "B": PRINT "SECOND"
40: END
```

Die folgenden Buchstaben und Zeichen können als Etikett verwendet werden: A, S, D, F, G, H, J, K, L, „, Z, X, C, V, B, N, M und SPC. Diese Tasten befinden sich in den beiden unteren Reihen der alphanumerischen Tastatur.

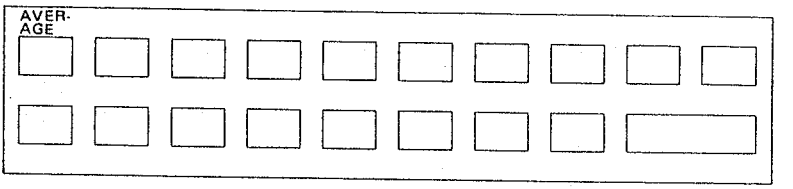
Hinweis: Zur Ausführung des Programms braucht anstelle von RUN 80 oder GOTO 10 nur die Taste [DEF] gedrückt und dann das Etikett eingegeben werden. Im obigen Beispiel würde durch Drücken von [DEF] und 'B' die Anzeige 'SECOND' erfolgen.

Bei Ausführung eines Programms mit DEF werden die Variable und Betriebsarteinstellung auf die gleiche Weise wie bie Verwendung von GOTO behandelt. Siehe Kapitel 10 für Einzelheiten.

7.2 TASTATUR-SCHABLONE

Mit dem Computer wird eine Schablone mitgeliefert, die sehr praktisch ist, um die Zuweisungen der Taste [DEF] zu kennzeichnen.
Wenn beispielsweise eine bestimmte Gruppe von Programmen häufig verwendet wird, können diese Programme etikettiert und entsprechend auf der Schablone gekennzeichnet werden. Wenn dann die Schablone über die beiden untersten Tastenreihen gesetzt wird, ist die Ausführung dieser Programme mit nur zwei Tastenbetätigungen sehr einfach.

Beispiel:



KAPITEL 8
VERWENDUNG VON DRUCKER UND KASSETTEN-INTERFACE

8.1 ALLGEMEINES

Die Option **SHARP CE-126P** (Thermodrucker mit integriertem Kassetten-Interface) ermöglicht es Ihnen, Ihre mit dem **PC-1425** erstellten Programme oder Daten auszudrucken bzw. mit einem externen Kassettenrecorder zu speichern.

Eigenschaften des **CE-126P:**

- 24-Zeichen-Thermodrucker
- Manueller oder programmgesteuerter Betrieb des Druckers
- 300-Baud-Kassetten-Interface
- Manueller oder programmgesteuerter Betrieb des Kassetten-Interface
- Trockenbatterie-Betrieb (Transportfähigkeit)/Netzadapterbetrieb

Der Anschluß des **CE-126P** an Ihren Computer ist in dem, dem **CE-126P** beigelegten Handbuch beschrieben.

8.2 VERWENDUNG ALS DRUCKER

Manueller Druckbetrieb (Protokollerstellung nur im RUN-Modus)

Der manuelle Druckbetrieb ermöglicht es Ihnen, Ihre Rechnungen sofort auf Papier zu protokollieren. Dieser Betrieb wird durch Drücken der [SHIFT] - und der [ENTER] -Taste eingeschaltet. Am Display wird dies durch einen kleinen waagerechten Strich oberhalb von 'PRINT' angezeigt.

Auf dem Thermopapier wird bei der Protokollführung die gleiche Zeichenfolge in derselben Form, wie am Display angezeigt, ausgedruckt.

Nach Abschluß der Eingabe mit [ENTER] wird der Druck gestartet. Vorher können Sie Ihre Eingaben, wenn notwendig, noch korrigieren.

Das folgende Bild zeigt eine Protokollführung als Beispiel:

```
12000*.065
                          780.
780+25.6
                         805.6
SIN 45
          7.071067812E-01
F=50
                           50.
X=2*π*F
                  314.1592654
P=60*X
                  18849.55592
P/32
                  589.0486225
```

Programmgesteuerter Ausdruck

Enthalten BASIC-Programme LPRINT-Anweisungen, so können die den Anweisungen folgenden Werte oder Textausdrücke über den angeschlossenen Thermodrucker (Option **CE-126P**) ausgedruckt werden.

Programme, die mit PRINT-Anweisungen geschrieben wurden, können ganz einfach umgewandelt werden, so daß die Ausgabe nicht über das Display, sondern über den angeschlossenen Drucker erfolgt. Dies wird mit dem Befehl PRINT = LPRINT erreicht, der in einer Programmzeile dem Programm vorangestellt oder im RUN-Mode direkt eingegeben werden kann.

Im letzteren Fall muß das Programm jedoch mit [DEF] -Taste oder GOTO < Zeilennummer > gestartet werden.

Mit dem Befehl PRINT = PRINT kann dieser Befehl wieder aufgehoben werden.

Die Umschaltung kann auch in Abhängigkeit von einem logischen Vergleichsausdruck innerhalb einer IF-Anweisung erfolgen.

Das im Hauptspeicher gespeicherte Programm kann mittels der LLIST-Anweisung am Drucker gelistet werden. Näheres hierzu siehe LLIST-Anweisung, Kapitel 10.

Ist die zu druckende Programmzeile länger als 24 Zeichen, so erfolgt die Ausgabe am Drucker zwei- bzw. mehrzeilig. Dabei wird der Druck ab der zweiten Zeile um vier bzw. um sechs Stellen eingerückt, um den Ausdruck übersichtlicher zu gestalten.

Hinweise:

1. Wenn während des Drucks ein Fehler auftritt, der im Zusammenhang mit nicht richtig transportiertem Papier steht, so reißen Sie das Papier an der Abrißkante ab und ziehen das verbleibende Papier aus dem Drucker.
   Löschen Sie die Fehlermeldung durch Drücken der [C·CE] -Taste und legen Sie die Papierrolle wieder richtig ein.
2. Wenn der Drucker durch irgendwelche externe elektrische Störfelder beeinflußt wird, kann es sein, daß der Ausdruck fehlerhaft ist. Drücken Sie die [BRK] -Taste, um den Druck zu unterbrechen. Schalten Sie den Thermodrucker **CE-126P** aus und nach einigen Sekunden wieder ein. Initialisieren Sie den Drucker durch Drücken von [C·CE].
3. Schalten Sie den Drucker nur ein, wenn Sie ihn benützen (Schonung der Batterien).

8.3 VERWENDUNG DES INTEGRIERTEN KASSETTEN-INTERFACES

Das integrierte Kassetten-Interface ermöglicht es Ihnen, Ihre Programme bzw. Daten auf Band zu speichern. Dazu benötigen Sie einen für Datenaufzeichnung geeigneten Kassettenrekorder, z.B. den **CE-152**.

Einmal gespeicherte Programme oder Daten können ganz leicht wieder in den Rechner geladen werden.

Auswahl des Kassettenrekorders

Wir empfehlen Ihnen, für Ihre Programmspeicherung den extra dafür vorgesehenen Kassettenrekorder CE-152 zu verwenden. Der CE-152 wurde eigens zur Speicherung der mit SHARP-Taschencomputern erstellten Programme oder Daten entwickelt.

Wenn Sie jedoch einen anderen Rekorder verwenden wollen, sollte dieser folgende Ausstattung haben:

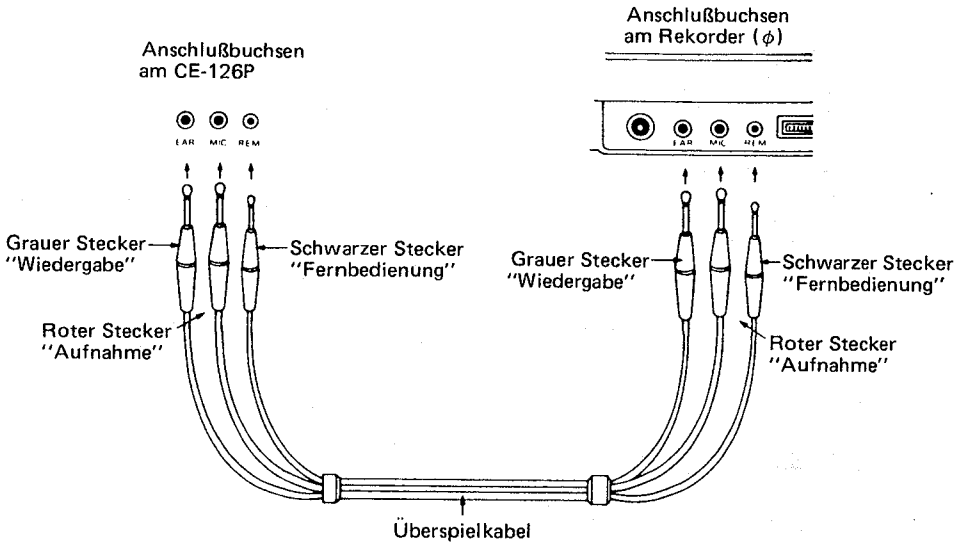
- Fernbedienung (REMOTE), Klinkenbuche 2,5 mm ϕ
- Bandzählwerk
- Mikrofoneingang (MIC), Klinkenbuchse 3,5 mm ϕ
- Ausgang (EAR), Klinkenbuchse 3,5 mm ϕ

Daneben sind folgende technische Daten gefordert:

– Eingangsimpedanz (MIC)	200. . .1000 Ohm
– Eingangsempfindlichkeit	Kleiner als 3 mV (–50 dB)
– Ausgangsimpedanz (EAR)	Kleiner als 10 Ohm
– Ausgangsleistung (EAR)	Größer als 1 V_{ss}
– Klirrfaktor	Kleiner als 15% im Bereich zwischen 2 . . . . 4 kHz
– Gleichlaufschwankungen	0,3% max. (W.R.M.S)

Anchluß des Kassettenrekorders an das CE-126P

Die herzustellenden Verbindungen sind in der folgenden Abbildung dargestellt.



8.4 DATENSPEICHERUNG AUF MAGNETBAND

Allgemeines

Auf dem Magnetband können Sie Daten und Programme abspeichern oder vom Band zurückladen. Außerdem können Sie das Magnetband als externen Programmspeicher verwenden, um große Programme ablaufen zu lassen.

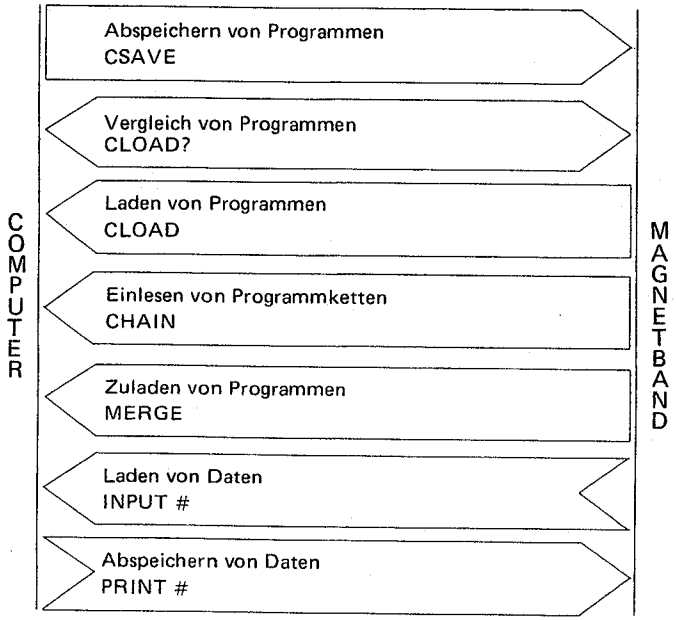
Die Informationen aus Daten und Programmen werden in Blöcken nach dem Zweitonverfahren auf das Magnetband geschrieben.

Auf einem Magnetband können abhängig vom Kassettentyp sehr viele Programme gespeichert werden. Damit Sie die Daten oder Programme wiederfinden und unterscheiden können, beginnt jeder Block aus einem Blocknamen (Programmnamen), der aus maximal 7 Zeichen besteht. Abgeschlossen wird der Speichervorgang, wenn die gesamte Information abgespeichert ist.

Um eine sichere Trennung der Blöcke zu gewährleisten, schreibt der Rechner vor dem Blocknamen automatisch etwa 5 bis 7 Sekunden lang einen konstanten Signalton. Ein kompletter Block hat damit folgende Form:

Signalton	Blockname	Programme oder Daten

Folgende Möglichkeiten werden bei der Datenspeicherung auf Magnetband geboten:



Hinweis:
Auf die Informationen auf Band können Sie nur sequentiell zugreifen. Damit Sie Ihre Daten und Programme schneller wiederfinden, sollten Sie sich vor dem Abspeichern Informationen über den Zählerstand des Kassettenrekorders und den Blocknamen der Daten aufschreiben. Sie werden sehr schnell eine umfangreiche Programmbibliothek von Daten erhalten.

Speichern von Daten und Programmen

Schießen Sie den Kassettenrecorder wie beschrieben an den CE-126P an. Nachdem Sie Ihr Programm eingegeben haben, können Sie den Rekorder für die Abspeicherung vorbereiten:

- Stellen Sie den REMOTE-Schalter des Interfaces auf OFF.
- Legen Sie eine Kassette in den Rekorder ein.
- Vergewissern Sie sich, daß die eingelegte Kassette zurückgespult ist.
- Suchen Sie eine freie Stelle am Band.
  Merken Sie sich den Zählerstand.
- Stellen Sie den REMOTE-Schalter des Interfaces auf ON.
- Bereiten Sie die Aufnahme vor:
  "RECORD"- und "PLAY"-Taste drücken. Lautstärkeregler auf Mitte bis Maximum stellen, Tonhöhenregler auf "Höhen".

– Kommando Eingabe:
  Mit der CSAVE-Anweisung können Sie Ihr Programm gleichzeitig mit einem Namen, dem Blocknamen (Programmnamen) versehen.

Wenn Sie die [ENTER] -Taste betätigt haben, setzt sich das Magnetband in Bewegung. Das Programm wird jetzt übertragen und unter dem angegebenen Namen abgespeichert. Zu Beginn hören Sie 5 bis 7 Sekunden den konstanten Signalton, anschließend eine Folge von Tönen. Sobald der Computer am Ende des Programms angelangt ist, stoppt das Band. Das Bereitschafftssymbol wird auf der Anzeige des Computers angezeigt.

Überprüfen der Abspeicherung

Nach der Abspeicherung Ihres Programms mit der CSAVE-Anweisung und bevor Sie das Programm im Computer löschen, ist es empfehlenswert, zu überprüfen, ob es fehlerfrei übertragen wurde. Das Band könnte beispielsweise eine schadhafte Stelle haben.

Die Überprüfung ist ganz einfach und erfolgt mit der CLOAD?-Anweisung. Der Computer vergleicht das Programm in seinem Speicher mit den auf dem Band gespeicherten Informationen. Ist die Aufzeichnung fehlerfrei, wird nach Abschluß der Überprüfung am Display des Computers wieder das Bereitschaftszeichen angezeigt.
Stellt der Computer beim Vergleich der Informationen einen Fehler fest, erfolgt die Fehlermeldung ERROR 8.
Löschen Sie das aufgezeichnete Programm und versuchen Sie es nochmals zu speichern. Gegebenenfalls verwenden Sie eine andere Bandstelle bzw. eine andere Kassette.

Laden von Programmen oder Daten

Beim Laden von Programmen gehen Sie wie folgt vor:

– Lautstärkeregler auf Mitte bis Maximum stellen; Tonhöhenregler auf "Höhen". Sollte der Kassettenrekorder bei Einstellung des Lautstärkereglers auf Maximum nicht ordnungsgemäß arbeiten, verringern Sie die Lautstärke und probieren erneut.
– Spulen Sie das Band an den Anfang zurück.
– Stellen Sie das Zählwerk auf Null.
– Suchen Sie mit Hilfe des Zählwerks die Bandstelle, an der das Programm beginnt.
– Schalten Sie den REMOTE-Schalter des Kassetten-Interface auf ON.
– Drücken Sie die "PLAY"-Taste des Kassettenrekorders.
– Geben Sie ein:

CLOAD "PROG. 1"

Wenn Sie die [ENTER] -Taste gedrückt haben, setzt sich das Magnetband in Bewegung. Sie hören die gleiche Signalfolge wie bei der Abspeicherung. Am Display wird während des Ladevorgangs ein Asterix (*) angezeigt.

Bei der Eingabe der CLOAD-Anweisung gibt es zwei Möglichkeiten:

CLOAD oder CLOAD "Programmname"

Dabei wird das Programm vom Magnetband in den Speicher des Rechners übertragen. Die beiden CLOAD-Anweisungen unterscheiden sich nur dadurch, daß bei der zweiten Anweisung der Blockname (Programmname) zusätzlich mit eingegeben wird. Damit wird erreicht, daß nur jenes Programm, dessen Blockname mit dem eingegebenen übereinstimmt, übertragen wird. Alle anderen auf Band gespeicherten Programme werden überlesen.

8.5 GRUNDLAGEN SHARP CE-140P/515P

Durch Anschließen des gesondert erhältlichen Farbmatrixdruckers CE-140P oder des Farbplotters CE-515P an den Computer haben Sie die Möglichkeit, Programme und Rechenergebnisse sowie Grafiken mehrfarbig auszudrucken. Der CE-140P kann eine Darstellung in sieben verschiedenen Farben zeichnen, wofür vier Farbkassetten gebraucht werden, während der CE-515P eine Tabelle oder ein Diagramm mit vier verschiedenen Farbstiften zeichnet. Der Drucker CE-140P kann direkt an die serielle I/O-Schnittstelle des Computers angeschlossen werden. Für den CE-515P hingegen ist ein Kabel (CE-516L) zum Anschluß an die serielle E/A-Schnittstelle erforderlich.

Schnittstelle und Drucker (Sonderzubehör)

- CE-126P Drucker/Cassettenrecorder-Schnittstelle
- CE-140P Farb-Matrixdrucker
- CE-515P/CE-516P Farb-Plotter/Drucker

Hinweise zum Gebrauch von Farbdruckern

Die BASIC-Befehle dieses Computers beinhalten verschiedene Druckerbefehle für Farb-Drucker bzw. Farb-Plotter sowie die Befehle LLIST und LPRINT zum Ausdrucken von Programmen oder Rechenergebnissen.
Der Farb-Plotter/Drucker CE-515P arbeitet nur dann, wenn sich die serielle E/A-Schnittstelle des Computer im "offenen" Zustand befindet. Vor der Ausgabe von Druckerbefehlen an den Plotter/Drucker muß daher zunächst der Befehl "OPEN" ausgeführt werden.

Druckbereich und Koordinaten für grafische Ausdrucke

Beim Ausdruck von Daten in grafischer Form mit einem Farb-Matrixdrucker bzw. Farb-Plotter/Drucker kann eine Druckerfunktion eingesetzt werden, die unter der Bezeichnung "Scissoring" bekannt ist. Mit dieser Ausschnitt-Funktion läßt sich ein gewünschter Teil einer grafischen Darstellung tatsächlich ausdrucken, während der nicht gewünschte bzw. außerhalb des Druckbereichs liegende Teil imaginär gedruckt, d.h. nicht zu Papier gebracht wird. Diese Funktion ist äußerst hilfreich zur Erstellung von Auszügen aus grafischen Darstellungen. Programme zum Ausdruck gesamter grafischer Darstellungen sind jedoch wesentlich leichter zu erstellen, so daß es sich bei sehr großen Darstellungen empfiehlt, diese in Übereinstimmung mit dem Papierformat in mehrere Abschnitte unterteilt auszudrucken.
Außerdem kann es vorkommen, daß bei Programmierfehlern der zu druckende Teil nur imaginär verarbeitet und nicht zu Papier gebracht wird. Beim Einsatz der Ausschnitt-Funktion (Scissoring) muß daher auf korrekte Programmierung geachtet werden.
Richtungen und Positionen zum Zeichnen von Linien werden in XY-Koordinaten ausgedrückt. "−" bezeichnet hierbei die Linksrichtung für X-Koordinaten (Horizontalrichtung), "+" die Rechtsrichtung.
Bei Y-Koordinaten (Vertikalrichtung) wird die Auf-Richtung durch "+", die Ab-Richtung durch "−" gekennzeichnet.
Weitere Einzelheiten zum Einsatz von Druckern sind der Bedienungsanleitung des Druckers zu entnehmen.
Wenn die Ausgabe auf den Drucker durch Drücken der Taste [BRK] unterbrochen wird, können bedeutungslose Zeichen auf dem Display erscheinen.

8.6 SERIELLE E/A-FUNKTIONEN

Der Computer ist mit einer seriellen E/A-Schnittstelle für Eingabe/Ausgabe versehen. Über diese Schnittstelle lassen sich die Farbdrucker CE-140P oder CE-515P an den Computer anschließen. Dies ermöglicht Grafikdruck in mehreren Farben und die Übertragung von Daten zwischen dem Computer und einer zentralen Datenverarbeitungsanlage.

Achtung: Vor dem Anschluß sollten die Spannungswerte überprüft werden, da eine Überspannung interne Bauteile des Computers zerstören könnte.

ALLGEMEINE HANDHABUNG DES SERIELLEN E/A-INTERFACE

Normalerweise ist das serielle Interface geschlossen. Dann können natürlich weder Daten empfangen noch Daten gesendet werden. Deshalb ist der erste Schritt des Datenaustausches das Öffnen des Kanals mit dem OPEN-Kommando. Sollte der Kanal allerdings schon geöffnet sein, so meldet der Rechner bei Ausführung des OPEN-Kommandos "ERROR 8".

Weiterhin ist es nötig, die Bedingungen für den Datentransfer festzulegen. Anders ausgedrückt: Die Form der Daten muß sowohl dem sendenden Computer als auch dem angeschlossenen Computer bekannt sein. Erwartet der empfangende Computer die Daten in einer anderen Form als der sendende sie herausgeschickt hat, so werden die Daten verfälscht. Die Bedingungen werden im OPEN-Kommando festgelegt.

Nachdem nun also der Kanal geöffnet ist und sowohl der sendende als auch empfangende Computer die Form der Daten kennt, können Daten mit den folgenden Kommandos ausgetauscht werden:

LPRINT, LLIST, SAVE, LOAD, PRINT #1, INPUT #1

Nach dem Ende des E/A-Programms wird der Kanal automatisch geschlossen. Auch das RUN-Kommando schließt den E/A-Kanal. In einem Programm kann man den Kanal mit dem CLOSE-Kommando schließen.

In einem Programm, das das serielle E/A-Interface benutzt, muß der Kanal also erst geöffnet werden, dann kann man Daten übertragen, und dann sollte der Kanal wieder geschlossen werden, alles wie oben beschrieben.

Achtung: Das Interface des Computers wartet nicht, ob angeschlossene Geräte zum Empfang bereit sind und hat auch keine Zeitfunktion, die den Übertragungsprozeß bei nicht betriebsbereiter Peripherie abschaltet. Die Datenübertragung wird also auch bei zum Beispiel abgeschaltetem Empfänger versucht, da der Computer die Ausführung nicht selbständig unterbrechen kann. In diesem Fall drücken Sie die BRK -Taste, um den Rechner zu stoppen.

Serielle E/A-Funktion bei Verwendung des CE-140P

Der Anschluß des Computers an den Farb-Punktmatrixdrucker CE-140P (Option) ermöglicht Ausdrucken an jedem Ort mit einer einzigen, kompakten Einheit.
An der Seite des Druckers CE-140P befindet sich eine serielle E/A-Schnittstelle zum Anschließen der Computer/Drucker-Einheit an einen Personal Computer.

* Der CE-140P Schalter sollte normalerweise auf SIO stehen. Abhängig von der Schalterstellung kann der Drucker wie folgt eingesetzt werden.

Schalterstellung des CE-140P	Serielle E/A-Schnittstelle	nur CE-140P ist angeschlossen	CE-140P und CE-126P sind angeschlossen*1
SIO (Normalerweise auf diese Position stellen.)	OFFEN	Eingabe/Ausgabe von CE-140P E/A-Schnittstelle	Gleich wie links
	GESCHLOSSEN	Ausgabe, gedruckt auf CE-140P	Ausgabe, gedruckt auf CE-126P*3
PRINTER	OFFEN*2	Ausgabe, gedruckt auf CE-140P	Gleich wie links
	GESCHLOSSEN	Ausgabe, gedruckt auf CE-140P	Ausgabe, gedruckt auf CE-126P*3

*1 Der Thermodrucker CE-126P (Option) wird an den Anschluß an der linken Seite des Computers zusammen mit dem Drucker CE-140P angeschlossen.

*2 Zur Verwendung dieser Konfiguration die Bedienungen für die serielle Eingabe/Ausgabe mit dem OPEN-Kommando spezifizieren, wie unten beschrieben. (Einstellung bei der Initialisierung: siehe unter OPEN-Kommando.)

OPEN "1200, N, 8, 1, A, C, &1A"
Wenn die E/A-Einstellung nicht korrekt ist, arbeitet der Drucker CE-140P nicht ordnungsgemäß.

*3 Kommandos für grafische Darstellungen werden auf dem CE-140P ausgegeben.

Achtung:

1. Vor Anschluß des CE-140P an den Computer zuerst die Stromzufuhr zum Computer abschalten.
2. Wenn die Anzeige für schwachen Batteriestrom (LOW BATTERY) am CE-140P aufleuchtet, reagiert der CE-140P nicht richtig auf Druckbefehle vom Computer, und es erscheinen Fehlermeldungen (Fehlercode 8).
3. Wenn die Tinte für den CE-140P ausgewechselt werden muß, kann die Stromversorgung des Computers nicht abgeschaltet werden, selbst wenn man den Stromschalter auf OFF schiebt. In diesem Fall zuerst die Taste — am CE-140P drücken und dann die Stromzufuhr des Computers abschalten.

KAPITEL 9
DIE RAM-KARTE

Der Computer speichert alle Programme und Daten (feste Variable, einfache Variable, Feldvariable usw.) in der RAM-Karte (Direktzugriffsspeicher). Die RAM-Karte CE-212M (8KByte) ist auch als Sonderzubehör erhältlich, jedoch bereits als Standardzubehör in Ihrem Computer installiert. Sechs weitere RAM-Karten werden als Sonderzubehör angeboten und können auf ganz einfache Weise in den Computer eingesetzt werden. Diese RAM-Karten bieten Speicherkapazitäten von 2KBytes bis 32 KBytes.
Durch die Verwendung einer RAM-Karte größerer Kapazität kann der Speicherbereich schnell und einfach erweitert werden. Da die RAM-Karten gegeneinander ausgetauscht werden können, können Porgramme, die gegenwärtig nicht erforderlich sind, gespeichert werden.
Das in den RAM-Karten gespeicherte Programm bleibt auch nach dem Herausnehmen der Karte erhalten, da die Karte mit einer eigenen Batterie zur Speicheraufrechterhaltung ausgerüstet ist. Damit kann ein Programm durch Einsetzen der entsprechenden Karte wiederverwendet werden.

Zusätzlich erhältliche RAM-Karten

Kartengröße	Typ	Kapazität	Programm/ Datenbereich	Zeitspanne der Daten-Aufrecht-erhaltung
1/2	CE-210M	2K Bytes	734 Bytes	40 Monate
	CE-211M	4K Bytes	2782 Bytes	24 Monate
	CE-212M	8K Bytes	6878 Bytes	15 Monate
	CE-2H16M	16 KBytes	15070 Bytes	14 Monate
Standard	CE-201M	8K Byets	6878 Bytes	34 Monate
	CE-202M	16KBytes	15070 Bytes	18 Monate
	CE-203M	32KBytes	31454 Bytes	18 Monate

ACHTUNGEN:
1. Karten der 1/2 Größe sind nur halb so groß wie Standardkarten.
2. RAM-Karten können auch als reine Daten- oder Systemdatenspeicher verwendet.
3. Die Zeitspanne der Daten-Aufrechterhaltung beschreibt die durchschnittliche Zeitspanne, während der die Inhalte der aus dem Computer herausgenommenen Ram-Karten aufrechterhalten bleibt.

9.1 AUSTAUSCH VON RAM-KARTEN

- Die RAM-Karte nur aus dem Computer nehmen, wenn sie gegen eine andere Karte ausgetauscht werden soll oder ihre Batterie ausgewechselt werden muß.
- Nach dem Einsetzen einer leeren Karte oder nach dem Austausch der Batterie einer RAM-Karte sind folgende Schritte unbedingt auszuführen:
  Den Computer austauschen und die ALL-RESET- Schalter drücken, wonach folgende Meldung auf dem Display erscheint.

BUSY
MEMORY ALL CLEAR O.K.?

Dann [ENTER], [Y] oder [=] drücken. Damit werden alle vorheirgen Inhalte der RAM-Karte gelöscht.

* Die Entnahme der Batterie aus der RAM-Karte löscht die in der Karte enthaltenen Daten und Programme. Deshalb sollten wertvolle Daten und Programme auf der Karte vor dem Batteriewechsel auf Band gespeichert werden.

Falls die Batterie von RAM-Karten der Standardgröße ausgewechselt werden, während diese Karten im Computer installiert sind, werden die Inhalte der Karte nicht gelöscht.

ACHTUNG: Während dem Einsetzen der RAM-Karte niemals den ALL-RESET-Schalter auf der Rückseite des Computers drücken.

Die Programm und Daten auf der RAM-Karte werden gelöscht, wenn der ALL-RESET-Schalter gedrückt wird. Falls es nötig werden sollten, den Rückstellschalter zu drükken, lesen Sie dazu bitte die Einzelheiten auf Seite 14 nach. Drücken Sie dann den Rückstellschalter, während die entsprechende Taste gedrückt gehalten wird.

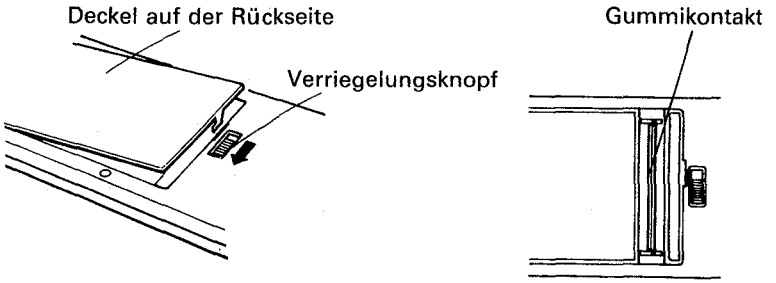
9.2 AUSTAUSCH VON RAM-KARTEN DER 1/2 GRÖßE

Falls die RAM-Karte zum ersten Mal verwendet werden soll, müssen Sie zuerst die Batterie in die RAM-Karte einsetzen.

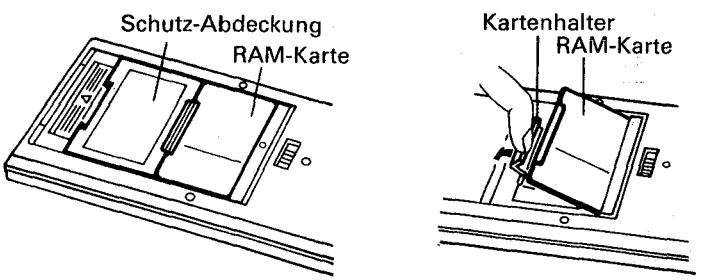
(1) Den Computer ausschalten.

(2) Die Arretierung des Deckels auf der Rückseite in Pfeilrichtung schieben und den Deckeln abnehmen.

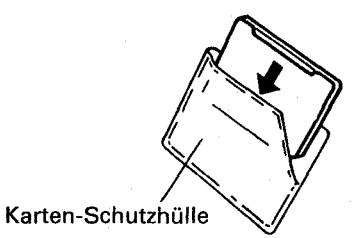
ACHTUNG: Die Schaltungen, Komponenten, Gummikontakte im Inneren nicht berühren.



(3) Den Kartendeckel entfernen und die RAM-Karte entnehmen, während Sie die Haltefeder zurückklappen.

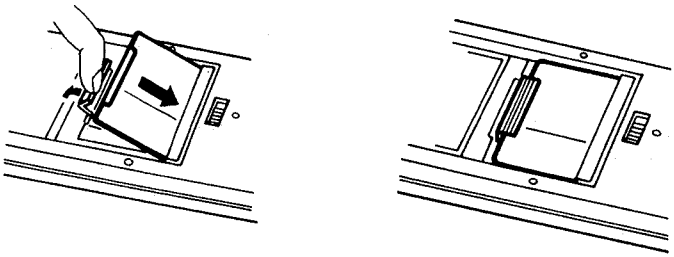


(4) Eine andere RAM-Karte aus deren Schutzhülle ziehen und die entnommene Karte in deren Schutzhülle geben.



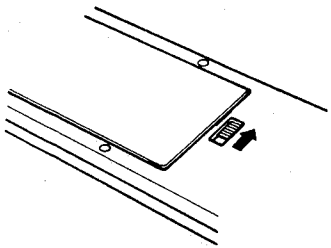
ACHTUNG: Wenn eine Karte der 1/2 Größe gegen eine der Standardgröße ausgetauscht wird, die Karte der halben Größe in die Schutzhülle der Standardgrößenkarte geben.

(5) Die Haltefeder anheben und die RAM-Karte mit nach vorne weisenden Kontakten in den Kartenschlitz einführen. Die Karte schräg ansetzen, damit die Kontakte richtig in den Schlitz greifen. Dann die Karte einlegen. Die Haltefeder loslassen.



ACHTUNG: Die Karte nicht anders herum einführen. Falsch eingelegte Karten können Kurzschlüsse und dadurch ernsthafte Schäden an Computer und Karte verursachen.

(6) Die RAM-Karte mit dem Deckel abdecken und den Rückwanddeckel wieder auf den Computer aufsetzen. Dann den Verriegelungsknopf in die Arretierungsposition schieben.



ACHTUNG: Unbedingt den Verriegelungsknopf in die Arretierposition schieben, da sonst der Computer nicht arbeiten kann. Falls der Computer bei geöffnetem Verriegelungsknopf eingeschaltet wurde, ist der Verriegelungsknopf zu schließen (LOCK-Position) und der Computer erneut einzuschalten.

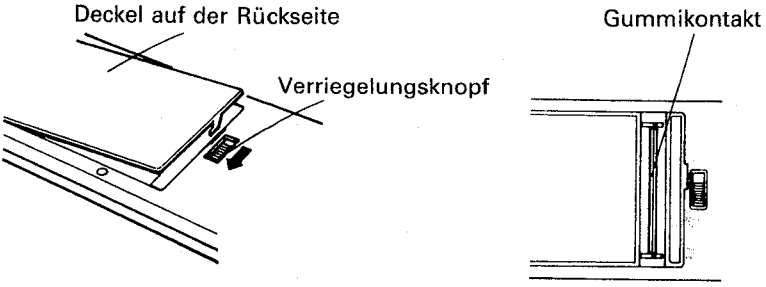
9.3 AUSTAUSCH VON RAM-KARTEN DER STANDARDGRÖßE

Falls die RAM-Karte zum ersten Mal verwendet werden soll, müssen Sie zuerst die Batterie in die RAM-Karte einsetzen.

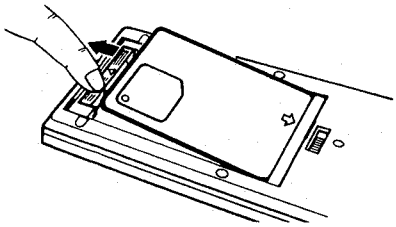
(1) Den Computer ausschalten.

(2) Die Arretierung des Deckels auf der Rückseite in Pfeilrichtung schieben und den Deckeln abnehmen.

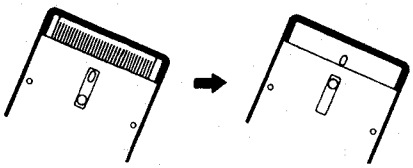
ACHTUNG: Die Schaltungen, Komponenten, Gummikontakte im Inneren nicht berühren.



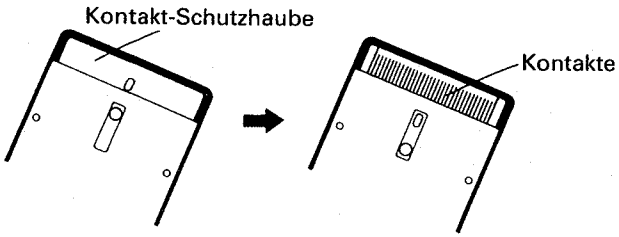
(3) Den Kartenhalter entsprechend der Abbildung in Pfeilrichtung ziehen. Damit wird die Karte freigegeben und kann entnommen werden.



(4) Die Kontakthaube nach Entnahme der RAM-Karte schließen.
ACHTUNG: Die Kontakthaube unbedingt ganz schließen.

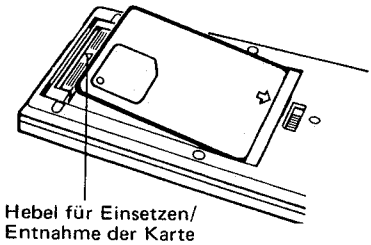


(5) Eine andere RAM-Karte aus deren Schutzhülle nehmen und die Schutzhaube von den Kontakten zurückziehen.



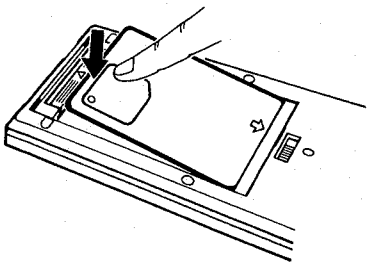
ACHTUNG: Die Kontakt-Schutzhaube ganz öffnen (die Kontakte liegen vollständig frei), jedoch die Kontakte niemals berühren.

(6) Die Karte entsprechend der Abbildung mit nach vorne weisenden Kontakten in den Computer einsetzen.



ACHTUNG: Die RAM-Karte niemals verkehrt herum oder mit verdeckten Kontakten einsetzen, da dies zu Kurschlüssen und damit zu ernsten Schäden an Computer und RAM-Karte führen kann.

(7) Dann den Batteriedeckel sanft aufsetzen. Dadurch rastet die Karte in den Kartenhalter ein.



(8) Den Rückwanddeckel aufsetzen und den Verriegelungsknopf in die Arretierungsstellung bringen.

ACHTUNG: Unbedingt den Verriegelungsknopf in die Arretierposition schieben, da sonst der Computer nicht arbeiten kann. Falls der Computer bei geöffnetem Verriegelungsknopf eingeschaltet wurde, ist der Verriegelungsknopf zu schließen (LOCK-Positon) und der Computer erneut einzuschalten.

9.4 VERWENDUNG DER RAM-KARTE MIT ANDEREN MODELLEN

Die RAM-Karte für diesen Computer kann auch mit dem PC-1350, PC-1450, PC-2500, PC-1100 oder PC-1460 verwendet werden. Deren RAM-Karten lassen sich ebenso mit diesem Computer benutzen.

(1) Bei der Verwendung von nicht-kompatiblen Programmen
Nach dem Einsetzen der RAM-Karte den Computer einschalten und dann den Rückstellschalter (RESET) auf der Rückseite des Computer alleine betätigen.

(2) Bei der Verwendung kompatibler Programme

1) Programme, die mit anderen Modellen erstellt wurden
Diese Programme können direkt benutzt werden. Jedoch auf folgende Punkte achten:

* Einige Änderungen sind erforderlich, falls das Programm Befehle verwendet, die mit diesem Computer nicht gegeben werden können.

* Zuvor in den Computer eingegebene Daten können für die Ausführung des Programms nicht verwendet werden, da sie gelöscht werden.

* Folgende Meldung erscheint auf dem Display, falls das Programm die Kapazität des Computers übersteigt.

BUSY
MEMORY ALL CLEAR O.K.?

Bei Erscheinen dieser Meldung schalten Sie den Computer einfach aus. Das Programm bleibt erhalten.
Zum Löschen des Programms drücken Sie [ENTER], [Y] oder [=] während dieser Meldung.
(Der Computer muß in der RAM-Karte einen Daten- und einen Systembereich reservieren können. Sollte dies nicht möglich sein, taucht die obige Meldung auf).
Falls innerhalb von 2 Minuten nach Erscheinen der "CLEAR O.K.?" Meldung keine Taste betätigt wird, schaltet sich der Computer automatisch aus.

2) Programme, die mit diesem Computer erstellt wurden
Bei installierter RAM-Karte die CONVERT ENTER Operation in der RUN-Betriebsart ausführen. Damit werden die auf der RAM-Karte befindlichen Programme automatisch konvertiert, wodurch sie von anderen Modellen ausgeführt werden können. Das Display erlischt nach abgeschlossener Umwandlung. (Während der Umwandlung niemals die BRK-Taste betätigen, da Sie den Umwandlungsvorgang sonst erneut beginnen müssen.) Beachten Sie bitte, daß die CE-203M Karte nur für diesen Computer vorgesehen ist. Falls Sie daher Programme auf dieser Karte umzuwandeln versuchen, wird eine Fehlermeldung ausgelöst.

* Programme mit Befehlen, die von PC-1350, PC-1450, PC- 2500 oder PC-1100 nicht ausgeführt werden können, müssen modifizert werden.

* Daten, die zuvor auf anderen Modellen eingegeben wurden, können nicht verwendet werden.

KAPITEL 10
BASIC REFERENZTEIL

Das folgende Kapitel ist in drei Teile untergliedert:

KOMMANDOS Instruktionen an den Rechnner, die außerhalb eines Programms gegeben werden, um Arbeitsbedingungen, Zusatzschaltungen und Programmkontrollen zu bestimmen.

BEFEHLE Kommando- und Befehlswörter, die gebraucht werden, um ein Programm zu erstellen.

FUNKTIONEN Spezielle BASIC-Operatoren, um Variable umzuwandeln.

Kommandos, die das serielle I/O-Interface und "TEXT" betreffen, werden auf folgenden Seiten erklärt:

Grafikbezogene Befehle . . . . . Seite 201-214
Verteilungsfunktionen . . . . . Seite 215-219
Befehle für das serielle I/O-Interface . . . . . Seite 231-247
Textfunktionen . . . . . Seite 248-250

Kommandos und verbale Befehle sind innerhalb jeder Kategorie in den betreffenden Abschnitten alphabetisch geordnet. Jeder Eintrag ist auf einer getrennten Seite, um das Auffinden zu erleichtern. Funktionen sind in vier Kategorien unterteilt und innerhalb jeder Kategorie alphabetisch geordnet. Der Inhalt von jedem Abschnitt ist auf den folgenden drei Seiten aufgeführt, damit Sie schnell die jeweilige Kategorie identifizieren und nachschlagen können.

KOMMANDOS

Programmkontrolle

CONT	NEW
DELETE	RENUM
GOTO*	RUN

Variablenkontrolle

CLEAR*
DIM*
ERASE*
MEM*

Cassettenkontrolle

CLOAD	INPUT#*
CLOAD?	MERGE
CSAVE	PRINT#*

Winkelmodus

DEGREE*	RADIAN*
GRAD*	

Debug-Kontrolle

LIST
LLIST
TROFF*
TRON*

Andere

BEEP*
PASS
RANDOM*
USING*
WAIT*

Grafikbezogene Befehle

COLOR*
CROTATE*
CSIZE*
GLCURSOR*
GRAPH*
LF*
LLINE*
LTEXT*
RLINE*
SORGN*

Serieller E/A-Anschluß

CLOSE#1
CONSOLE
INPUT#1
LLIST
LOAD
LPRINT
OPEN
OPEN$
PRINT#1
SAVE

* Diese Kommandos sind auch BASIC-Befehle. Ihre Wirkung als Kommando ist indentisch mit ihrer Wirkung als Befehl. Sie sind beschrieben in dem Abschnitt BEFEHLE.

BEFEHLE

Kontrollen und Verzweigungen

CHAIN
END
FOR . . . TO . . . STEP
GOSUB
GOTO
IF . . . THEN
NEXT
ON . . . GOSUB
ON . . . GOTO
RETURN
STOP

Eingabe und Ausgabe

AREAD
CSAVE
DATA
INPUT
INPUT#
INPUT#1
LOAD
LPRINT
PAUSE
PRINT
PRINT#
PRINT#1
READ
RESTORE
USING
WAIT

Zuweisung und Deklarationen

CLEAR
DIM
ERASE
LET

Andere

BEEP
CLOSE#1
CONSOLE
DEGREE
GRAD
OPEN
RADIAN
RANDOM
REM
TROFF
TRON

Grafikbezogene Befehle

COLOR
CROTATE
CSIZE
GLCURSOR
GRAPH
LF
LLINE
LTEXT
RLINE
SORGN

FUNKTIONEN

Pseudovariable

INKEY$
MEM
PI

String-Funktionen

ASC
CHR$
LEFT$
LEN
MID$
RIGHT$
STR$
VAL

Text-Funktionen

BASIC
TEXT

Numerische Funktionen

ABS
ACS
ASN
ATN
COS
CUR
EXP
FACT
INT
LN
LOG
RCP
RND
ROT
SGN
SIN
SQR
SQU
TAN
TEN

Verteilungsfunktionen

BDS
FDS
NDS
PDS
PFD
PND
PTD
PXD
TDS
XDS

10.1 KOMMANDOS

CLOAD

1. CLOAD
2. CLOAD "dateiname"

Abkürzung: CLO., CLOA.
Vergleiche: CLOAD?, CSAVE, MERGE, PASS

Wirkung:
Mit dem CLOAD-Kommando werden Programme von der Cassette in den Rechner geladen.

Anwendung:
Die erste Form des CLOAD-Kommandos löscht den Speicher des Rechners und lädt das erste Programm von der eingelegten Cassette.

Die zweite Form des Kommandos löscht den Speicher und lädt das mit "dateiname" angegebene Programm von der Cassette.

Ist der Computer im PROgramm- oder RUN-Modus, so wird das Programm von der Cassette in den Programmspeicher geladen.

Beispiele:

CLOAD	Lädt das erste Programm von der Cassette.
CLOAD "PRO3"	Sucht auf der eingelegten Cassette das Programm PRO3 und lädt es ein.

Bemerkungen:
1. Wird die angegebene Datei nicht gefunden, sucht der Rechner weiter nach der Datei, selbst wenn die Cassette abgelaufen ist. Unterbrechen Sie in diesem Fall die Suche mit der BRK -Taste. Das bezieht sich auch auf die Kommandos MERGE, CHAIN, CLOAD? und INPUT#, die später beschrieben werden.
2. Tritt während der Ausführung der Kommandos CLOAD oder CHAIN (wird später beschrieben) ein Fehler auf, so ist das im Speicher vorhandene Programm defekt.

* Während des Ladevorgangs wird an der rechten Seite das Symbol "*" angezeigt. Dieses Symbol verschwindet, wenn das Laden beendet ist. Wird ein Dateiname gesucht, erscheint dieses Symbol erst, wenn der Ladevorgang beginnt.

CLOAD?

1. CLOAD?
2. CLOAD? **"dateiname"**

Abkürzung: CLO.?, CLOA.?
Vergleiche: CLOAD, CSAVE, MERGE, PASS

Wirkung:
Mit dem Kommando CLOAD? vergleichen Sie das Programm im Speicher des Rechners mit einem auf Cassette gespeicherten Program.

Anwendung:
Um zu prüfen, ob ein Programm richtig gespeichert wurde, spulen Sie die Cassette an den Anfang zurück und geben das Kommando CLOAD? ein.

Die erste Form des CLOAD?-Kommados vergleicht den Speicherinhalt mit dem ersten auf der Cassette gefundenen Programm.

Die zweite Form des Kommandos CLOAD? sucht nach dem mit "dateiname" angegebenen Programm und vergleicht es dann mit dem Speicherinhalt.

Beispiele:

CLOAD?	Vergleicht den Speicherinhalt mit dem ersten auf der Cassette gefundenen Programm.
CLOAD? "PRO3"	Sucht auf der eingelegten Cassette nach dem Programm PRO3 und vergleicht es dann mit dem Speicherinhalt.

* Das Symbol "*" erscheint ganz rechts auf dem Display, während das Programm überprüft wird. Am Ende des Prüfvorgangs verschwindet das Sternchen, und es erscheint das Bereitschaftssymbol.

CONT

1. CONT

Abkürzung: C., CO., CON.
Vergleiche: RUN, STOP-Befehl

Wirkung:
Mit dem CONT-Kommando setzen Sie die Ausführung eines unterbrochenen Programms fort.

Anwendung:
Wurde die Ausführung eines Programms mit dem Befehl STOP oder mit der Taste BRK unterbrochen, so kann es mit dem Kommando CONT fortgesetzt werden.

CONT können Sie auch benutzen, wenn die Ausführung des Programms zeitweise beispielsweise durch Kommandos wie PRINT unterbrochen wurde.

Beispiel:

CONT	Setzt die Ausführung eines unterbrochenen Programms fort.

CSAVE

1. CSAVE
2. CSAVE **"dateiname"**
3. CSAVE, **"passwort"**
4. CSAVE **"dateiname"**, **"passwort"**

Abkürzung: CS., CSA., CSAV.
Vergleiche: CLOAD, CLOAD?, MERGE, PASS

Wirkung:
Mit diesem Kommando wird ein Programm aus dem Speicher des Rechners auf der Cassette abgespeichert.

Anwendung:
Die erste Form des Kommandos CSAVE speichert das Programm im Speicher des Rechners ohne speziellen Dateinamen auf der Cassette ab.

Die zweite Form des Kommandos weist dem Programm vor Abspeicherung den angegebenen Dateinamen zu und speichert es unter diesem Namen auf der Cassette ab.

Die dritte Form des Kommandos speichert das Programm ohne Dateinamen, aber mit einem Passwort auf der Cassette ab. Programme, die mit einem Passwort geschützt sind, können von jedem geladen und ausgeführt werden. Gelistet oder verändert werden kann es aber nur dann, wenn das korrekte Passwort eingegeben wird (näheres unter der Beschreibung des PASS-Kommandos).

Mit der vierten Form des CSAVE-Kommandos wird den Programm sowohl ein Dateiname als auch ein Passwort zugewiesen.

Beispiel:

CSAVE "PRO3", "GEHEIM"	Speichert das präsente Programm unter dem Dateinamen PRO3 mit dem Passwort GEHEIM auf Cassette ab.

DELETE

1. DELETE **[nr. der ersten zeile]** **[,[nr, der letzten zeile]]**

Abkürzung: DEL., DELE., DELET.
Vergleiche: NEW, PASS

Wirkung:

Das DELETE-Kommando dient zum Löschen einer bzw. mehrerer Programmzeilen. Dieser Befehl wird zur manuellen Operation in der Programmierbetriebsart verwendet.

Anwendung:

Zum Löschen mehrerer chronologisch aufeinanderfolgender Programmzeilen wird die Nr. der ersten und letzten zu löschenden Zeile angegeben.
Wenn eine der beiden Zeilennummern nicht auffindbar ist, tritt ein Fehler auf.

Wenn nur die Nr. der ersten Zeile angegeben wird, erfolgt Löschung dieser einen Zeile.

Wenn die Nr. der ersten Zeite mit nachfolgendem Komma (,) angegeben wird, wird diese Zeile zusammen mit allen nachfolgenden Zeilen gelöscht.

Wenn das Komma (,) mit nachfolgender Nr. der letzten Zeile angegeben wird, werden alle Zeilen ab Programmanfang einschließlich der angegebenen Zeilennummer gelöscht.

Wenn beide Zeilennummern ausgelassen werden, tritt ein Fehler auf.

Bei mit MERGE geladenen Programmen wirkt DELETE nur auf das zuletzt geladene Programm.

Wenn das Programm durch ein Kennwort geschützt ist, wird das DELETE-Kommando ignoriet.

Beispiel:

DELETE 1ØØ, Löscht Zeile 100 und alle nachfolgenden Zahlen.

GOTO

1. GOTO **ausdruck**

Abkürzung: G., GO., GOT.
Vergleiche: RUN

Wirkung:
Mit dem GOTO-Kommando wird die Ausführung eines Programms begonnen.

Anwendung:
Das GOTO-Kommando kann genauso wie das RUN-Kommando benutzt werden. Die Ausführung eines Programms wird von der durch ausdruck angegebenen Zeilennummer gestartet.

GOTO unterscheidet sich in sechs Punkten vom RUN-Kommando:

1. Der Wert des WAIT-Intervalles wird nicht zurückgesetzt.
2. Wurde die Anzeige durch einen USING-Befehl formatiert, wird sie nicht gelöscht.
3. Werte von Variablen und Feldern (Arrays) bleiben erhalten.
4. PRINT=LPRINT-Status wird nicht zurückgesetzt.
5. Der READ-Zeiger wird nicht zurückgesetzt.
6. Das serielle I/O Interface bleibt aktiv.

Beispiel:

GOTO 100 Die Ausführung eines Programmes wird mit Zeile 100 gestartet.

LIST

1. LIST
2. LIST **zeilennummer**
3. LIST **"label"**

Abkürung: L., LI., LIS.
Vergleiche: LLIST

Wirkung:
Mit dem LIST-Kommando wird ein Programm zur Anzeige gebracht.

Anwendung:
Das LIST-Kommando kann nur im PROgramm-Modus benutzt werden.

* Mit LIST wird die erste Programmzeile angezeigt.
* Mit LIST **zeilennummer** wird das Programm mit der angegebenen Zeilennumer angezeigt. Gibt es die angegebene Zeilennumer nicht, so wird das Programm mit der nächsthöheren Zeilennummer angezeigt.
* Mit LIST **"label"** wird das Programm mit der mit diesem Etikett versehenen Zeile angezeigt.
* Sind mit dem MERGE-Kommando mehrere Programme in den Speicher geladen, so wird mit dem LIST-Kommando die erste Zeilennummer des zuletzt geladenen Programms gelistet.
* Wird das Kommando in der Form LIST "label" benutzt und das angegebene Label im zuletzt geladenenen Programm nicht gefunden, so sucht das Kommando das Label in den anderen Programmen im Speicher. Die dazugehörige Zeile wird dann gelistet.

  Bei einem durch ein Passwort geschützten Programm wird das LIST-Kommando nicht ausgeführt.

Beispiel:

LIST 100 Die Zeile 100 wird angezeigt.

LLIST

1. LLIST
2. LLIST **ausdruck**
3. LLIST **ausdruck 1, ausdruck 2**
4. LLIST **ausdruck,**
5. LLIST, **ausdruck**
6. LLIST **"label"**

Abkürzung: LL., LLI., LLIS.
Vergleiche: LIST

Wirkung:
Mit dem LLIST-Kommando kann ein Programm auf den Druckern CE-126P bzw. CE-140P ausgegeben (gelistet) werden.

Anwendung:
* Wenn das serielle I/O-Interface durch das OPEN-Kommando aktiviert wurde, erfolgt die Ausgabe auf einem angeschlossenen Terminal. Das CLOSE-Kommando desaktiviert das serielle I/O-Interface.

Das LLIST-Kommando kann im PROgramm-Modus und im RUN-Modus benutzt werden.

Die erste Form druckt das gesamte Programm in Speicher aus. Die zweite Form druckt nur die mit **ausdruck** angegebene Zeile aus. Die dritte Form des Kommandos druckt das Programm von der durch den **ausdruck1** angegebenen Zeile oder der nächstgrößeren bis zu der durch den **ausdruck2** angegebenen Zeile oder der nächstgrößeren aus. Zwischen den beiden angegebenen Zeilen müssen mindestens zwei Zeilen liegen. Die vierte Form listet alle Programme ab der durch **ausdruck** eingegebenen Zeile bis zum Ende. Die fünfte Form des Kommandos LLIST druckt die Programmzeilen bis zur eingegebenen Zeile (inklusive) aus.

Wenn mit dem MERGE-Kommando mehrere Programme eingeladen wurden, so wirkt das LLIST-Kommando auf das zuletzt geladene Programm. Um ein zuvor geladenes Programm aufzurufen, benutzen Sie das Kommando LLIST "label".

Ein gesetztes Passwort unterdrückt das LLIST-Kommando.

* Zum Ausdrucken eines Programms auf dem Drucker CE-140P die Anzahl der Zeichen pro Zeile auf 24 oder mehr einstellen. (Siehe das CONSOLE-Kom-

mando.) Wenn die Anzahl der Zeichen pro Zeile auf weniger als 24 eingestellt ist und der Befehl LLIST ausgeführt wird, tritt ein Fehler (ERROR 3) auf.

* Beim Ausdrucken von Programmen auf dem Drucker CE-140P wird "$\sqrt{\ }$" als SQR ausgedruckt.

Beispiel:

LLIST 10Ø, 2ØØ — Die Programmzeillen zwischen 10Ø und 2ØØ werden ausgedruckt.

MERGE

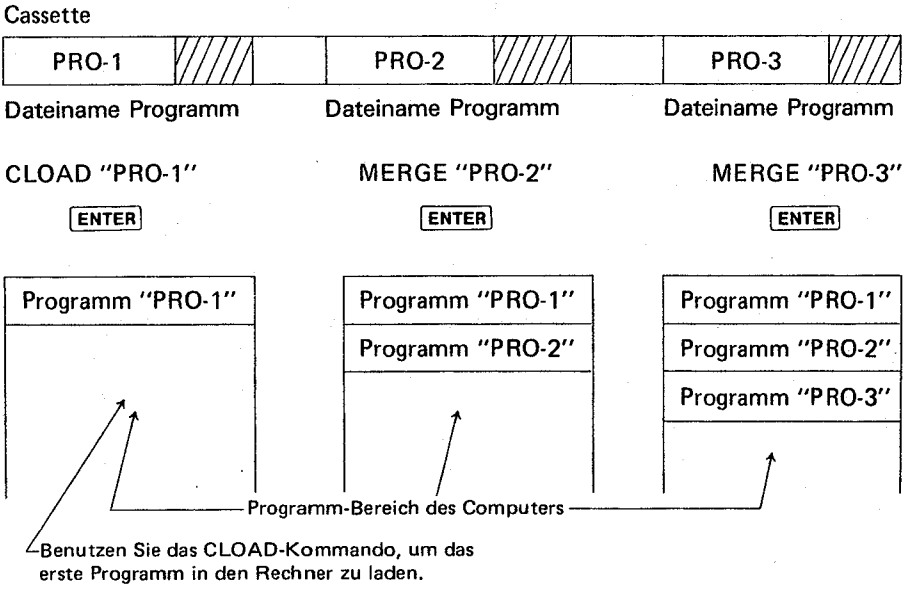
1. MERGE
2. MERGE "dateiname"

Abkürzung: MER., MERG.
Vergleiche: CLOAD

Wirkung:
Mit dem MERGE-Kommando können Programme von der Cassette an bereits im Speicher vorhandene angeknüpft werden.

Anwendung:
Mit dem MERGE-Kommando wird das bereits im Speicher geladene Programm aufbewahrt und ein weiteres Programm von der Cassette in den Speicher des Rechners geladen. Auf diese Weise können mehrere Programme in den Computer geladen werden.

Beispiele:
Nehmen wir an, wir haben drei Programme mit den Dateinamen PRO1, PRO2 und PRO3 auf Cassette abgespeichert. Jetzt kann das Programm PRO1 mit dem Kommando CLOAD eingeladen werden, die Programme PRO2 und PRO3 mit dem MERGE-Kommando. Die Abbildung auf dieser Seite verdeutlicht, wie die Programme gespeichert werden.



Programme, die mit dem MERGE-Kommando geladen wurden, werden wie im o. g. Beispiel gespeichert.

* Ist die erste Zeilennummer des mit dem MERGE-Kommando geladenen Programms größer als die letzte Zeilennummer des vorher geladenen Programms, werden die beiden Programme im folgenden wie eins behandelt.

* Ist die erste Zeilennummer des mit dem MERGE-Kommando geladenen Programms kleiner als die letzte Zeilennummer des vorher geladenen Programms, werden die beiden Programme als zwei verschiedene behandelt.
  In dem obigen Beispiel sind die Zeilennummern für die ausgewählten Programme PRO1, PRO2 und PRO3 10–200, 50–150 und 160–300. Die Programme PRO1 und PRO2 werden als zwei verschiedene Programme behandelt. PRO2 und PRO3 werden bei den Zeilennummern 50–300 als ein Programm behandelt.

* Wenn Sie Programme mit dem MERGE-Kommando laden, kann es zwei oder mehrere Programme mit den gleichen Zeilennummern geben. In diesem Fall beziehen sich die Kommandos RUN oder GOTO in der Form RUN ausdruck bzw. GOTO ausdruck auf das letzte mit MERGE geladene Programm. Dann können die zuvor geladenen Programme nicht mehr ausgeführt werden.
  Geben Sie deshalb immer ein Label am Beginn eines Programms an und lassen Sie es anschließend ausführen.
  Beachten Sie, daß in jedem Fall nur das zuletzt mit MERGE geladene Programm editiert werden kann, wenn Sie das MERGE-Kommando geben. Alle zuvor mit MERGE geladenen Programme können nicht editiert werden. Geben Sie deshalb immer das Label eines Programms an, ehe Sie mit dem MERGE-Kommando das nächste Programm laden.

VERKNÜPFUNG PASSWORT-GESCHÜTZTER PROGRAMME

Wollen Sie ein Passwort-geschütztes Programm mit dem MERGE-Kommando laden, so müssen wir zwei Fälle unterscheiden:

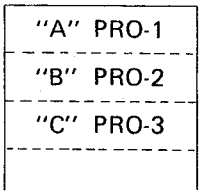
1. Das Programm im Speicher ist geschützt.
2. Das Programm im Speicher ist nicht geschützt.

Im ersten Fall kann kein Passwort-geschütztes Programm mit MERGE geladen werden. Im zweiten Fall werden alle Programme im Speicher durch das neu geladene, geschützte Programm ebenfalls zu Passwort-geschützten Programmen.

Sind im Speicher geschützte Programme und wird ein ungeschütztes Programm mit MERGE geladen, so wird dieses neu geladene Programm ebenfalls zu einem Passwort-geschützten Programm.

AUSFÜHRUNG MIT MERGE GELADENER PROGRAMME

Die Abbildung zeigt die Speicherorganisation, nachdem PRO1 geladen wurde. Anschließend wurde mit dem MERGE-Kommando PRO2 und PRO3 geladen. Wenn Sie RUN oder GOTO (RUN ausdruck, GOTO ausdruck) benutzen, um ein Programm zu starten, so wird immer PRO3 ausgeführt.



Benutzen Sie andererseits RUN "label" oder GOTO "label" oder eine andere definierte Taste, um den Ladevorgang zu starten, wird der Rechner nach dem angegebenen Label im Programm PRO3 suchen.

Wird das Label dort nicht gefunden, so setzt der Rechner seine Suche in dem Programm PRO1 fort. Ist die Suche auch dort nicht erfolgreich, wird PRO2 auf das gesuchte Label hin überprüft. Findet der Rechner das gewünschte Label, wird das Programm von der mit dem Label bezeichneten Zeile abgearbeitet.

Wenn der Rechner auf diese Weise ein Label sucht und das gewünschte Label in Programm PRO3 auftaucht, können die Programme PRO1 und PRO2 nicht ausgeführt werden.

NEW

1. NEW

Wirkung:

Das NEW-Kommando löscht Programme und Daten im Programmspeicher.

Anwendung:

Wird es im PROgramm-Modus benutzt, so werden alle Programme und Daten gelöscht. (Mit Passwort geschützte Programme können nicht gelöscht werden.)

Das NEW-Kommando ist im RUN-Modus nicht definiert. Sie erhalten nach Eingabe des NEW-Kommandos die Fehlermeldung ERROR 9.

Beispiel:

NEW Löscht den entsprechenden Speicherbereich.

PASS

1. PASS **"zeichenkette"**

Abkürzung: PA., PAS.
Vergleiche: CSAVE, CLOAD, NEW

Wirkung:
Mit dem Kommando PASS können Passwörter gesetzt und gelöscht werden.

Anwendung:
Mit einem Passwort schützen Sie Ihr Programm vor Einsichtnahme oder Modifikation durch andere Anwender. Ein Passwort besteht aus einer Zeichenkette bis zu sieben Zeichen. Die erlaubten Zeichen für das Passwort sind alle Buchstaben des Alphabets und folgende Sonderzeichen:

```
!  #  $  %  &  (  )  *
+  –  /  .  ,  :  ;  <
=  >  ?  @  ^  √  %
```

Wurde das Kommando PASS gegeben, so ist das Programm im Speicher geschützt. Das bedeutet, daß das Programm nicht mehr editiert werden kann. Es kann nicht auf Cassette abgespeichert oder mit LIST oder LLIST ausgegeben werden. Weiterhin können weder Zeilen hinzugefügt noch gelöscht werden. Die einzige Möglichkeit, das Passwort auszuschalten, ist ein weiteres PASS-Kommando mit dem gleichen Passwort einzugeben.

Achtung: Wenn Sie ein Passwort mit mehr als 7 Zeichen eingeben, werden nur die ersten 7 Zeichen als gültig erkannt und dem Programm zugewiesen.

Geben Sie [ENTER] ein, nachdem Sie das Passwort bestimmt haben.

Sie erhalten eine Fehlermeldung, wenn Sie nach der Eingabe des Passwortes weitere Zeichen oder Symbole eingeben. In einem solchen Fall akzeptiert der Rechner das Passwort nicht. Es erfolgt eine Fehlermeldung nach Eingabe von PASS "ABCDEFG": A = 123 [ENTER] .

Beispiel:

PASS "GEHEIM" Setzt für alle Programme im Speicher das Passwort "GEHEIM".

* Wenn das Passwort gesetzt wurde und die RAM-Karte entnommen wird, ist das in der RAM-Karte gespeicherte Programm geschützt.

RENUM

1. RENUM **[neue Zeilennummer][,[alte Zeilennummer][,Inkrement]]**

Abkürzung: REN., RENU.

Wirkung:
Der RENUM-Kommand dient zur Neunumerierung von Programmzeilen. Dieser Befehl ist wirksam für manuellen Betrieb in der PRO-(Programm-)Betriebsart.

Anwendung:
Dieser Befehl numeriert in den festgelegten inkrementalen Stufen alte Zeilennummern neu. Dieser Befehl ist wirksam für manuellen Betrieb in der PRO-(Programm) Betriebsart.

Bei Auslassung von Werten der neuen Zeilennummer und Inkrement wird für beide 10 angenommen. Wird die alte Zeilennummer ausgelassen, beginnt die Neunumerierung von der ersten Zeile des Programms. Kann die festgelegte Zeilennummer nicht gefunden werden, tritt eine Fehlermeldung ein.

Beispiel 1:
RENUM
Alle Programmzeilen in Inkrementen von 10 Stufen ab Zeile 10 neu numerieren.

Beispiel 2
RENUM 1ØØ, 5Ø, 1Ø
Alte Zeilennummer 50 in neue Zeilennummer 100 ändern und nachfolgende Zeilennummern in Inkrementen von 10 Schritten neu numerieren.

Das RENUM-Kommando ändert automatisch alle Zeilennummern-Bezugnahmen, die auf GOTO, GOSUB, IF THEN, ON GOTO, ON GOSUB, RESTORE usw. folgen, um die neuen Nummeren wiederzugeben. In diesem Falle tritt jedoch ein Fehler ein, wenn eine der Zeilennummern im Programm eine Variable (z.B. GOTO A) oder ein Ausdruck (z.B. GOTO 2*50) ist. Tritt ein Fehler aufgrund einer solchen, falschen Zeilennummer-Bezugnahme ein, kann eine Neunumerierung der betreffenden Zeilennummer nicht durch RENUM ausgeführt werden. In einem solchen Fall den eine unkorrekte Zeilennummer zu einer REM-Anweisung enthaltenden Befehl vorübergehend neu schreiben und nach Ablauf des RENUM-Kommand korrigieren (eventuell ändern in ON GOTO).

Das RENUM-Kommando kann nicht ausgeführt werden, wenn die Anzahl der neu zu numerierenden Zeilen 65279 überschreitet oder wenn die Angabe eine Änderung in der Ablauffolge der Programmzeilen verlangt (zum Beispiel: Es wird der Versuch unternommen, RENUM 15, 30 bei drei bestehenden Programmzeilen 10, 20 und 30 auszuführen.)

Der Ablauf von RENUM nimmt bei einem großen Programm etwas Zeit in Anspruch. Wird, während ein Sternchen (*) am äußersten rechten Ende des Displays erscheint, die Taste [BRK] zur Unterbrechung des Programms gedrückt, kehrt das Programm vor beendetem Ablauf von RENUM in den Ausgangszustand zurück. Erscheinen jedoch zwei Sternchen (**) auf dem Display, wird die Unterbrechung durch die [BRK] -Taste ignoriert.
Der Arbeitsbereich (Anzahl der Programmzeilen × 4) von Bytes wird nur bei Ablauf des RENUM-Kommandos benutzt. Durch Neunumerieren von Programmzeilen ändern sich die Zeilennummern-Bezugnahmen durch GOTO, GOSUB usw. ebenfalls. Das Originalprogramm darf deshalb eine erhöhte Anzahl an verwendeten Bytes aufweisen. Mit anderen Worten verwendet die neue Zeile GOTO 200 ein Byte mehr als die alte Zeile GOTO 20. Das RENUM-Kommando kann nicht ausgeführt werden, wenn die verbleibende Kapazitaut des Arbeitsbereichs aufgrund der höheren Byteanzahl reduziert wird. In einem solchen Fall sind Variablen durch das CLEAR-Kommando aus dem Speicher zu löschen, wonach eine Ausführung von RENUM eventuell möglich ist.

(Siehe Anhang A bezüglich der durch RENUM verursachten Fehlermeldungen.)

RUN

1. RUN
2. RUN **zeilennummer**

Abkürzung: R., RU.
Vergleiche: GOTO, MERGE

Wirkung:
Mit dem RUN-Kommando wird das Programm im Speicher gestartet.

Anwendung:
Die erste Form des Kommandos startet das Programm im Speicher mit der Zeile, die die niedrigste Zeilennummer hat.

Die zweite Form startet das Programm mit der angegebenen Zeilennummer. Die Kommandos RUN und GOTO unterscheiden sich in sechs Punkten.

* Wenn mehrere Programme mit dem MERGE-Kommando geladen werden, wird das zuletzt geladene Programm mit dem Kommando RUN oder mit dem Kommando RUN ausdruck ausgeführt.

Für das Kommando RUN gelten folgende Merkmale:

1. Der Wert des WAIT-Intervalls wird zurückgesetzt.
2. Eine durch einen USING-Befehl erzeugte Formatierung der Anzeige wird gelöscht.
3. Variable und Felder (Arrays) werden gelöscht.
4. PRINT = PRINT-Status wird gesetzt.
5. Der READ-Zeiger wird auf den Anfang der ersten DATA-Zeile gesetzt.
6. Schließt den seriellen E/A-Anschluß (serieller Anschluß).

Die Ausführung eines Programms mit GOTO und mit der [DEF] -Taste sind gleich. Alle drei Formen des Programmstarts löschen FOR/NEXT- und GOTO-Gruppen.

Beispiel:

RUN 100 Startet ein Programm mit der Zeile 100.

10.2 BEFEHLE

AREAD

1. AREAD **variable**

Abkürzung: A., AR., ARE., AREA.
Vergleiche: Befehl INPUT und die Beschreibung der DEF -Taste in Kapitel 7.

Anwendung:
Der AREAD-Befehl liest den vor dem Programmstart auf der Anzeige stehenden Wert in eine Variable ein.

Wirkung:
Wenn einem Programm ein Label (ein "Name") in Form eines Buchstabens zugewiesen wurde, so daß es mit der DEF -Taste gestartet werden kann, kann der AREAD-Befehl benutzt werden, um einen einzelnen Startwert einzugeben, ohne den INPUT-Befehl zu setzen. Der Befehl AREAD muß in der ersten Zeile nach der Zeile mit dem Label gesetzt sein. Taucht er an anderer Stelle im Programm auf, so wird er ignoriert. Es können sowohl Zeichenketten (Strings) als auch numerische Werte eingegeben werden, aber immer nur ein Wert pro Programm.

Erstellen Sie Ihr Programm im PROgramm-Modus. Der AREAD-Befehl kommt dann im RUN-Modus zum Tragen. Geben Sie erst den gewünschten Wert ein und dann drücken Sie erst DEF , gefolgt von dem Label, das Sie gesetzt haben. Wird ein String benutzt, so muß dieser nicht in Anführungsstriche gesetzt werden.

Beispiele:

```
10 "X": AREAD N
20 PRINT N ^ 2
30 END
```

Erstellen Sie sich dieses Beispiel-Programm im PROgramm-Modus.

Die Eingabe

7 DEF X

gibt das Quadrat der Zahl "7", also "49" aus.

Wichtig:

1. Steht vor Beginn der Programmausführung das Bereitschaftssymbol (>) in der Anzeige, so wird die verwendete Variable auf Ø (Null) bzw. auf ASCII-Leerzeichen gesetzt.
2. Wird zu Beginn des Programms mit dem PRINT-Befehl etwas angezeigt, so wird folgendes gespeichert:

Folgendes Programm soll ausgeführt werden:

```
1Ø "A": PRINT "ABC", "DEFG"
2Ø "S": AREAD A$: PRINT A$
RUN Modus
DEF A ⟶ ABC  DEFG
DEF S ⟶ DEFG
```

* Wird

  PRINT numerischer ausdruck, numerischer ausdruck

  oder

  PRINT string, string

  ausgegeben, so wird der letzte angezeigte Ausdruck gespeichert.

* Wird

  PRINT numerischer ausdruck; numerischer ausdruck; numerischer ausdruck; . . .

  ausgegeben, so wird der erste angezeigte Ausdruck gespeichert.

* Wird

  PRINT *string; string; string; . . .*

  ausgegeben, so wird der letzte angezeigte Ausdruck gespeichert.

BEEP

1. BEEP **ausdruck**

Abkürzung: B., BE., BEE.

Wirkung:
Mit dem BEEP-Befehl wird ein Ton erzeugt.

Anwendung:
Der Computer erzeugt auf den BEEP-Befehl hin einen oder mehrere 4 kHz-Töne. Die Anzahl der Töne wird mit **ausdruck** angegeben (positive Zahlen kleiner als 9.999999999 E 99). Nur der ganzzahlige Anteil des angegebenen Ausdrucks wird im BEEP-Befehl verwertet.

BEEP kann auch mit numerischen Variablen arbeiten oder als Kommando benutzt werden. In diesem Falle werden die Töne direkt nach dem [ENTER] erzeugt.

Beispiele:

```
10 A = 5 : B$ = "9"
20 BEEP 3          Erzeugt 3 Töne.
30 BEEP A          Erzeugt 5 Töne.
40 BEEP (A+4)/2    Erzeugt 4 Töne.
50 BEEP B$         Nicht erlaubt, ERROR 9 wird ausgegeben.
60 BEEP -4         Es wird kein Ton erzeugt, aber auch keine Fehlermeldung
                   ausgegeben.
```

CHAIN

1. CHAIN
2. CHAIN **ausdruck**
3. CHAIN **"dateiname"**
4. CHAIN **"dateiname", ausdruck**

Abkürzung: CHA., CHAI.
Vergleiche: CLOAD, CSAVE, RUN

Wirkung:
Mit dem CHAIN-Befehl wird ein auf Cassette gespeichertes Programm ausgeführt. Er kann nur in Verbindung mit den Optionen CE-126P und CE-152 benutzt werden.

Anwendung:
Um den CHAIN-Befehl benutzen zu können, muß mindestens ein Programm auf Cassette abgespeichert sein. Dieses Programm wird dann mit dem CHAIN-Befehl geladen und ausgeführt.

Die erste Form des Befehls lädt das erste gefundene Programm von Cassette und beginnt die Ausführung mit der niedrigsten Zeile dieses Programms. Der Effekt ist der gleiche wie mit dem CLOAD- und dem RUN-Kommando im RUN-Modus.

Die zweite Form lädt das erste gefundene Programm und beginnt die Ausführung mit der durch **ausdruck** angegebenen Zeile.

Die dritte Form von CHAIN sucht das mit **"dateiname"** angegebene Programm, lädt es und führt es aus, beginnend mit der kleinsten Zeilennummer dieses Programms.

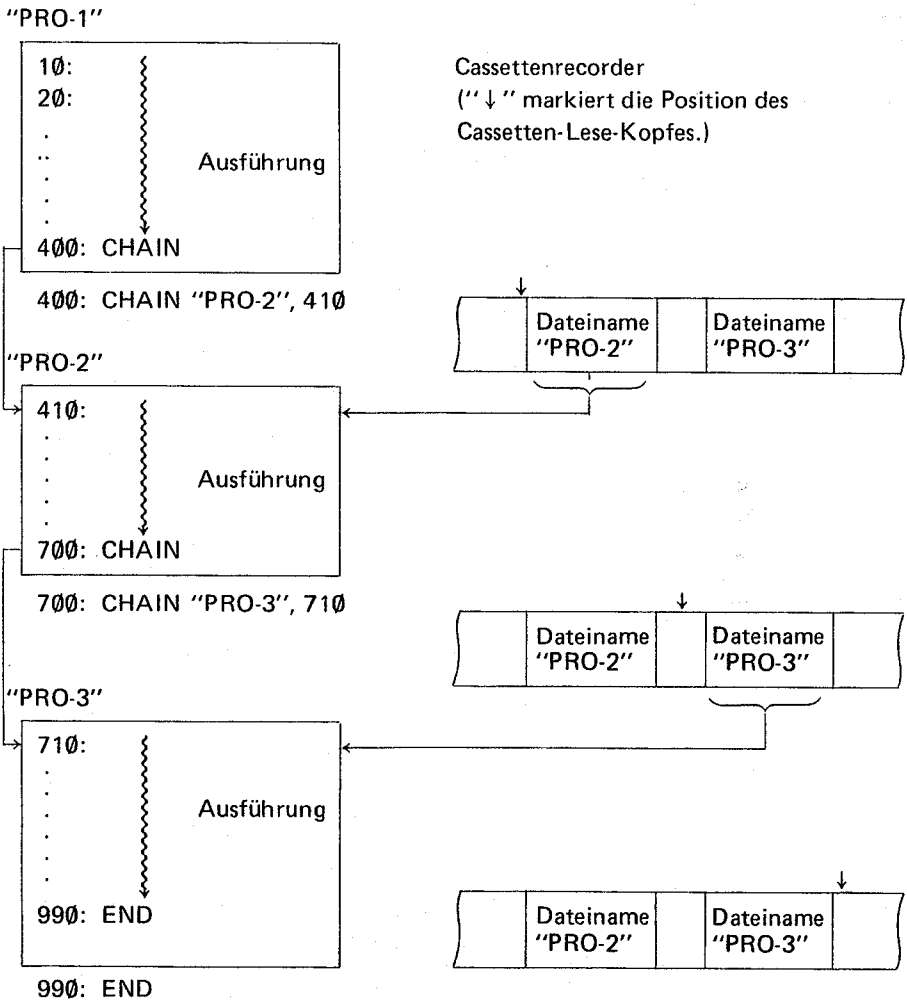
Die vierte Form des CHAIN-Befehls sucht das mit **"dateiname"** bezeichnete Programm, lädt es von Cassette und startet die Ausführung mit der in **ausdruck** angegebenen Zeile.

Beispiele:

1Ø CHAIN	Lädt das nächste Programm von Cassette und beginnt die Ausführung mit der niedrigsten Zeilennummer.
2Ø CHAIN "PRO2", 48Ø	Sucht auf der Cassette das Programm PRO2, lädt es und startet es mit Zeile 48Ø.

Nehmen Sie z.B. einmal an, daß Sie drei Programme mit dem Namen PRO1, PRO2 und PRO3 auf Cassette gespeichert haben. Alle Programme enden mit einem CHAIN-Befehl.

Während der Programmausführung wird der Rechner, wenn er auf einen CHAIN-Befehl trifft, die folgende Sektion in den Speicher rufen und sie ausführen. Auf diese Weise werden schließlich alle Sektionen abgearbeitet.



Achtung: Programme, die mit MERGE geladen wurden, sollten in der Regel keine CHAIN-Anweisungen enthalten.

CLEAR

1. CLEAR **[variable] [,variable,...,variable]**

Abkürzung: CL., CLE., CLEA.
Vergleiche: DIM

Wirkung:

Mit dem CLEAR-Befehl werden die Inhalte der festen Variable und die anderen Variable (z.B. einfache Variable) gelöscht.

Anwendung:

Der CLEAR-Befehl löscht die einfache Variable für den spezifizierten Variablennamen und setzt die festen Variablen auf 0 oder Null. Werden die Variablennamen ausgelassen, werden alle einfache Variable und Feldvariable gelöscht (auf ihren nicht definierten Zustand), die Inhalte der festen Variablen werden gelöscht.

Beispiele:

```
100 "A": CLEAR:DIM B(4)
200 "B": CLEAR A, AD, D1$
```

DATA

1. DATA **ausdruck, ausdruck, ausdruck, ausdruck . . .**

Abkürzung: DA., DAT.
Vergleiche: READ, RESTORE

Wirkung:
Der DATA-Befehl stellt für den READ-Befehl die benötigten Daten bereit.

Anwendung:
Soll ein Feld (Array) mit Daten gefüllt werden, empfiehlt es sich, diese Daten in DATA-Befehlen bereit zu stellen und diese Daten dann mit einer FOR/NEXT-Schleife und einem READ-Befehl in das Feld einzulesen. Der erste READ-Befehl liest den ersten Wert aus der ersten DATA-Zeile, der zweite READ-Befehl den zweiten Wert und so fort. Mit jedem READ-Befehl wird ein Zeiger auf den nächsten Wert gesetzt. Sind die Daten einer DATA Zeile erschöpft, so wird der Zeiger an den Anfang der nächsten DATA-Zeile gesetzt.

DATA-Befehle haben keinen Einfluß auf den Ablauf des Programms, so daß sie an jeder Stelle des Programms eingefügt werden können. Viele Programmierer setzen DATA-Zeilen gerne hinter die Zeile, in der der dazugehörende READ-Befehl steht. Andere setzen alle DATA-Zeilen an den Anfang oder an das Ende eines Programmes. Wenn nötig, kann der Zeiger mit dem RESTORE-Befehl an den Anfang einer bestimmten DATA-Zeile gesetzt werden. So ist es auch möglich, eine DATA-Zeile mehrmals zu benutzen.

Beispiele:

```
10: DIM B(10)
20: WAIT 128
30: FOR I=1 TO 10
40: READ B(I)
50: PRINT B(I)
60: NEXT I
70: DATA 10,20,30,40,50,60
80: DATA 70,80,90,100
90: END
```

Initialisierung eines Feldes

Lädt die Werte aus den DATA-Zeilen in das Feld B( ). B(1) wird gleich 10, B(2) gleich 20, B(3) gleich 30 gesetzt, usw.

DEGREE

1. DEGREE

Abkürzung: DE., DEG., DEGR., DEGRE.
Vergleiche: GRAD, RADIAN

Wirkung:
Mit dem DEGREE-Befehl wird der Winkel in der Einheit ALT-GRAD interpretiert.

Anwendung:
Der Computer kann drei Formen der Winkeldarstellung verarbeiten:

1. ALT-GRAD
2. BOGENMASS
3. NEU-GRAD

Diese Formen werden für die Darstellung der Werte der Argumente von SINus-, COSinus- und TANgens-Funktionen und der Ergebnisse der Umkehrfunktionen ArcusSiNus, ArcusCoSinus und ArcusTaNgens benutzt.

Beispiel:

```
10: DEGREE
20: X = ASN 1
30: PRINT X
```

X hat jetzt den Wert 90, dem Arcussinus von 1.

DIM

1. DIM **dim liste**

dim liste	**dimensionierung** oder **dimensionierung, dim liste**
dimensionierung	**numerische dim** oder **string dim**
numerische dim	**numerischer name (größe)**
string dim	**string name (größe)** oder **string name (größe) * länge**
numerischer name	**erlaubte numerische variable**
string name	**erlaubte string variable**
größe	**größe** oder **größe, größe**
größe	**zahl der elemente**
länge	**länge eines strings im array**

Abkürzung: D., DI.

Wirkung:
Der DIM-Befehl wird benutzt, um Speicher für numerische oder String-Felder frei zu halten.

Anwendung:
Alle Feldvariablen, ausgenommen die Form A( ), A$( ) und einfache Variablen (AA, B1 etc.), müssen mit einem DIM-Befehl initialisiert werden, um den nötigen Speicher zu reservieren.

Ein Feld kann maximal 2 Dimensionen haben. Die maximale Länge einer Dimension ist 255. Zu der angegebenen Zahl addiert sich ein "nulltes" Element, so daß die Anzahl der Elemente einer Dimension immer um eins größer ist als angegeben, z.B. hat das Feld DIM B(3) die Elemente B(Ø), B(1), B(2) und B(3). In zweidimensionalen Feldern haben beide Dimensionen ein "Null"-Element.

In String-Feldern kann zusätzlich zur Anzahl der Elemente die Länge der einzelnen Strings vorgegeben werden, z.B. reserviert DIM B$(3) * 12 Speicherraum für vier String-Elemente mit je 12 Zeichen Länge. Wird die Länge der Strings nicht vorgegeben, so wird eine maximale Länge von 16 Zeichen angenommen.

Bei der Initialisierung eines Feldes werden alle Werte

a) eines numerischen Feldes auf Ø (Null)
b) eines STRING-Feldes auf ASCII-Leerzeichen gesetzt

Informieren Sie sich hinsichtlich der Initialisierung und Dimensionierung der Felder A( ) und A$( ) in dem Abschnitt, in dem die Variablen erklärt werden.

Beispiele:

1Ø: DIM B(1Ø)	Reserviert Speicherraum für ein numerisches Feld mit 11 Elementen.
2Ø: DIM C$(4,4) * 1Ø	Reserviert Speicherraum für ein zweidimensionales String-Feld mit 5 Spalten und 5 Reihen. Jeder String soll maximal 1Ø Zeichen lang sein.

END

1. END

Abkürzung: E., EN.

Wirkung:
Der END-Befehl signalisiert das Ende eines Programms.

Anwendung:
Sind mehrere Programme geladen, so ist es nötig, den einzelnen Programmen Endmarken zu geben, damit bei der Ausführung eines Programms der Rechner nicht in ein anderes hineinläuft. Der Befehl END setzt diese Endmarke.

Der END-Befehl desaktiviert automatisch das serielle I/O-Interface und ersetzt das CLOSE-Kommando.

Beispiel:

```
10: PRINT "HALLO"
20: END
30: PRINT "TSCHUESS"
40: END
```

Wenn Sie das Kommando RUN 10 geben, so gibt der Rechner das Wort HALLO aus, aber nicht das Wort TSCHUESS. Dieses wird auf das Kommando RUN 30 ausgegeben.

ERASE

1. ERASE **feld** [,**feld**,...,**feld**]

Abkürzung: ER., ERA., ERAS.
Vergleiche: CLEAR

Wirkung:

Der ERASE-Befehl dient zum Löschen der Feldvariablen.

Anwendung:

Der ERASE-Befehl löscht die Feldvariablen für die spezifizierten Feldnamen von den mit dem DIM-Befehl deklarierten Feldvariablen. Die Feldvariablen für die Kalkulationstabellen-Daten (T0$( ), D0$( ), D0(,) usw.) können nur mit dem ERASE-Befehl gelöscht werden.
Wenn eine spezifizierte Feldvariable nicht vorhanden ist, erfolgt keine Fehlermeldung.

Beispiele:

ERASE AB, XY$ Löscht die Feldvariable AB(5) und XY$(2,4).

FOR . . . TO . . . STEP

1. FOR **numerische variable** = **ausdruck1** TO **ausdruck2**
2. FOR **numerische variable** = **ausdruck1** TO **ausdruck2** STEP **ausdruck3**

Abkürzung: F., FO. ; STE.
Vergleiche: NEXT

Wirkung:
Der FOR-Befehl wird in Verbindung mit dem NEXT-Befehl gegeben, um eine bestimmte Operation mehrmals zu wiederholen.

Anwendung:
Der FOR- und der NEXT-Befehl schließen eine Gruppe von Befehlen ein, die wiederholt werden sollen. Wird diese Gruppe das erste Mal ausgeführt, so hat die Schleifenvariable den Wert von **ausdruck1**.

Erreicht das Programm nun den NEXT-Befehl, so wird die Schleifenvariable um den **ausdruck3** erhöht und dann mit dem **ausdruck2** verglichen. Ist der Wert der Schleifenvariablen kleiner oder gleich dem **ausdruck2**, so wird die eingeschlossene Gruppe ein weiteres Mal ausgeführt, beginnend mit dem Befehl hinter FOR. In der ersten Form des Befehls ist die Erhöhung (STEP) gleich 1. In der zweiten Form wird die Erhöhung durch den **ausdruck3** vorgegeben. Ist der Wert der Schleifenvariablen größer geworden als **ausdruck2**, so wird das Programm mit dem Befehl hinter NEXT fortgesetzt. Da der Vergleich am Ende der FOR/NEXT-Schleife durchgeführt wird, werden die eingeschlossenen Befehle mindestens einmal ausgeführt.

Der Wert von **ausdruck1**, **ausdruck2** und **ausdruck3** muß zwischen –9.999999999 E 99 und 9.999999999 E 99 liegen. Wird **ausdruck3** gleich Ø gesetzt, so kann das Programm nicht fortgesetzt werden.

Die Schleifenvariable kann in der umschlossenen Gruppe von Befehlen benutzt werden. Mit ihr kann z. B. ein Feld-Index "hochgezählt" werden. Man sollte aber sehr vorsichtig sein, wenn der Wert der Schleifenvariablen verändert wird.

Programme dürfen niemals so geschrieben werden, daß von einem Befehl außerhalb einer FOR/NEXT-Schleife in eine solche hineingesprungen wird. Ein Programm sollte auch nicht aus einer FOR/NEXT-Schleife herausspringen. Sie sollten eine FOR/NEXT-Schleife immer über den NEXT-Befehl verlassen. Der Schleifenvariablenwert muß dazu nur größer als **ausdruck2** gesetzt werden.

Eine Gruppe von Befehlen, die von einer FOR/NEXT-Schleife eingefaßt werden, kann eine weitere FOR/NEXT-Schleife enthalten. Diese muß dann eine andere Schleifenvariable haben und vollständig in der ersten Schleife liegen. Wenn also in einer FOR/NEXT-Schleife ein FOR-Befehl auftaucht, so muß auch der dazugehörenden NEXT-Befehl in der Schleife liegen. Sie können bis zu fünf FOR/NEXT-Schleifen ineinanderlegen.

Beispiele:

```
10: FOR I=1 TO 5
20: PRINT I
30: NEXT I
```

Diese Schleife gibt die Zahlen 1, 2, 3, 4 und 5 aus.

```
40: FOR N=10 TO 0 STEP -1
50: PRINT N
60: NEXT N
```

Diese Schleife zählt rückwärts von 10 nach 0 und gibt die Schleifenvariable aus.

```
70: FOR N=1 TO 10
80: X=1
90: FOR F=1 TO N
100: X=X * F
110: NEXT F
120: PRINT X
130: NEXT N
```

Diese Schleife berechnet die Fakultäten der Zahlen von 1 bis 10 und gibt das Ergebnis auf der Anzeige wieder.

Achtung: Programme sollten nicht aus einer FOR/NEXT-Schleife herausspringen. Es könnte hierbei passieren, daß Sie eine Fehlermeldung (ERROR 5) im Programmablauf (bei Programmen, die mehrere FOR/NEXT-Schleifen enthalten) angezeigt bekommen.

GOSUB

1. GOSUB **ausdruck**

Abkürzung: GOS., GOSU.
Vergleiche: GOTO, ON . . . GOSUB, ON . . . GOTO, RETURN

Wirkung:
Mit dem GOSUB-Befehl verzweigt das Programm zu einer BASIC-Unterroutine.

Anwendung:
Wenn eine Gruppe von Befehlen in einem Programm an verschiedenen Stellen ausgeführt werden soll oder diese Befehlsgruppe in verschiedenen Programmen eingesetzt werden soll, so empfiehlt sich die Verwendung der Befehle GOSUB und RETURN.

Die Befehlsgruppe wird so untergebracht, daß das Programm normalerweise nicht in diese Befehle hineinlaufen kann. Eine Möglichkeit wäre, diese Gruppe hinter den END-Befehl zu stellen. An den Stellen des Hauptprogramms, an der diese Gruppe nun ausgeführt werden soll, fügen Sie den GOSUB-Befehl mit der Zeilennummer der Startzeile der Unterroutine als **ausdruck** ein. Die letzte Zeile der Unterroutine muß einen RETURN-Befehl enthalten. Wird der GOSUB-Befehl nun ausgeführt, so wird das Programm in der Unterroutine fortgesetzt. Erreicht der Conputer den RETURN-Befehl so wird die Programmausführung mit dem auf den GOSUB folgenden Befehl im Hauptprogramm fortgesetzt.

In einer Unterroutine kann ein GOSUB vorkommen. Maximal können 1Ø Unterroutinen ineinandergelegt werden.

Der **ausdruck** im GOSUB-Befehl darf kein Komma enthalten, z.B. A(2,5) kann nicht benutzt werden. Um von einem bestimmten Punkt mehrere Unterroutinen anspringen zu können, wird der Befehl ON. . . GOSUB benutzt. Der **ausdruck** kann nur eine erlaubte Zeilennummer sein, d. h. der Wert muß zwischen 1 und 65279 inklusive liegen oder Sie bekommen die Fehlermeldung ERROR 4.

Beispiel:

```
1Ø: GOSUB 1ØØ
2Ø: END
1ØØ: PRINT "HALLO"
11Ø: RETURN
```

Das Programm springt von Zeile 1Ø in die Unterroutine in Zeile 1ØØ, gibt das Wort HALLO aus und springt dann in die Zeile 2Ø des Hauptprogramms zurück.

GOTO

1. GOTO **ausdruck**

Abkürzung: G., GO., GOT.
Vergleiche: GOSUB, ON . . . GOSUB, ON . . . GOTO

Wirkung:
Mit dem GOTO-Befehl verzweigt sich das Programm.

Anwendung:
Der Befehl veranlaßt den Computer, von einer bestimmten Stelle in einem Programm an eine andere, durch den **ausdruck** bestimmte Stelle des Programms zu springen. Anders als im GOSUB-Befehl "erinnert" sich der Rechner nicht, von wo dieser Sprung ausgeführt wurde.

Der **ausdruck** des Befehls darf kein Komma wie z.B. in A(1,2) enthalten. Soll von einem Punkt die Möglichkeit gegeben werden, an eine von mehreren Stellen zu springen, so muß der ON . . . GOTO-Befehl benutzt werden. Der **ausdruck** muß eine Zeilennummer sein. Erlaubt sind die Werte von 1 bis 65279. Bei einem anderen Wert bekommen Sie die Fehlermeldung ERROR 4.

Gute Programme laufen von Anfang bis Ende ohne Sprünge, abgesehen von Verzweigungen zu Unterprogrammen. Der prinzipielle Gebrauch des GOTO-Befehls ist dashalb im IF . . . THEN-Befehl zu finden.

Beispiel:

```
1Ø: INPUT A$
2Ø: IF A$="J" THEN GOTO 5Ø
3Ø: PRINT "NEIN"
4Ø: GOTO 6Ø
5Ø: PRINT "JA"
6Ø: END
```

Wenn Sie ein "J" eingeben, gibt der Rechner das Wort JA aus, sonst das Wort NEIN.

GRAD

1. GRAD

Abkürzung: GR., GRA.
Vergleiche: DEGREE, RADIAN

Wirkung:
Mit dem GRAD-Befehl werden die Winkel in der Einheit NEU-GRAD interpretiert.

Anwendung:
Der Computer hat drei Formen der Winkeldarstellung:

1. ALT-GRAD
2. BOGENMASS
3. NEU-GRAD

Die Argumente der Funktionen SINus, COSinus und TANgens können in dieser Form eingegeben bzw. die Ergebnisse der Umkehrfunktionen ArcusSiNus, ArcusCoSinus und ArcusTaNgens in diesen Formen ausgegeben werden.

Beispiel:

```
10: GRAD
20: X=ASN 1
30: PRINT X
```

X hat jetzt den Wert 100, den Gradienten von Arcussinus 1.

IF . . . THEN

1. IF **bedingung** THEN **befehl**
2. IF **bedingung befehl**

Abkürzung: keine für IF ; T., TH., THE.

Wirkung:

Das IF . . . THEN-Befehlspaar wird benutzt, um einen bestimmten Befehl nur dann auszuführen, wenn eine Bedingung erfüllt ist.

Anwendung:

Im normalen Ablauf eines BASIC-Programms werden die Befehle in der Reihenfolge ihres Auftretens abgearbeitet. Das IF . . . THEN-Befehlspaar erlaubt es, einen Befehl nur dann auszuführen, wenn eine bestimmte Bedingung erfüllt ist. Ist der Bedingungsteil des IF-Befehls wahr, so wird der Befehl ausgeführt, ist die Bedingung falsch, wird er übersprungen.

Der Bedingungsteil des IF-Befehls kann jeder vergleichende Ausdruck sein. Es ist auch möglich, einen numerischen Ausdruck als Bedingung zu setzen, obgleich der Sinn des Befehls dadurch wenig klar wird. Jeder Ausdruck, der kleiner oder gleich Null ist, setzt die Bedingung falsch. Jeder positive Wert setzt die Bedingung wahr.

Der Befehl, der auf den THEN-Teil des Befehls folgt, kann jeder BASIC-Befehl sein, auch ein weiterer IF . . . THEN-Befehl. Handelt es sich um eine Zuweisung, so muß der LET-Befehl mit dem Wort LET eingeleitet werden.

Die beiden Formen des Befehls sind in der Ausführung gleich, aber die erste Form ist wesentlich klarer.

Beispiel:

```
10: INPUT "WEITER?"; A$
20: IF A$="JA" THEN GOTO 10
30: IF A$="NEIN" THEN GOTO 60
40: PRINT "JA ODER NEIN, BITTE"
50: GOTO 10
60: END
```

Dieses Programm wiederholt die Frage WEITER?, solange JA eingegeben wird. Es stoppt, wenn NEIN eingegeben wird. Andere Antworten werden nicht beachtet.

Achtung:

Der Ausdruck THEN darf keinesfalls in Ausdrücken ausgelassen werden, die mit den Worten LLINE, RLINE, GLCURSOR, LTEXT, GRAPH, LF, CSIZE, COLOR, SORGN oder CROTATE beginnen.

INPUT

1. INPUT **eingabe liste**

eingabe liste **ist eingabe gruppe**

oder

eingabe gruppe, eingabe liste

eingabe gruppe **ist var list**

oder

prompt, var liste

oder

prompt; var liste

var liste **ist variable**

oder

variable, var liste

prompt ist jede String-Konstante

Abkürzung: I., IN., INP., INPU.
Vergleiche: INPUT #, READ, PRINT

Wirkung:
Der INPUT-Befehl ermöglicht es, einen oder mehrere Werte über die Tastatur einzugeben.

Anwendung:
Wollen Sie in einem Programm die Grundwerte für jeden Programmlauf ändern, so empfiehlt sich der INPUT-Befehl. Die Basiswerte können dann mit der Tastatur eingegeben werden.

In der einfachsten Form fordert der Befehl Sie mit einem Fragezeichen in der linken Ecke der Anzeige auf, die nötige Eingabe zu machen. Sie geben dann den gewünschten Wert, gefolgt von [ENTER] ein. Dieser Wert wird dann der ersten Variablen der Variablenliste des INPUT-Befehls zugewiesen. Sind in dem Befehl mehrere Variable aufgelistet, so wird der Prozeß so oft wiederholt, bis alle Variablen belegt sind.

Sie können die Eingabe-Aufforderung (= Prompt), also das Fragezeichen, durch einen Prompt-String, das ist eine Bemerkung, die die nötige Eingabe erklären sollte, eigener Wahl ersetzen. Der Ablauf des Zuweisungsprozesses ändert sich nur insofern, als das Fragezeichen durch diesen String ersetzt wird.

Sollte der Befehl einen Prompt-String und die Variablenliste mehr als eine Variable enthalten, so wird der String nur bei der ersten Aufforderung gezeigt. Die folgenden werden wieder mit dem Fragezeichen gekennzeichnet. Beinhaltet der Befehl einen zweiten Prompt-String, so wird dieser vor der Zuweisung der direkt folgenden Variablen angezeigt.

Wenn Sie bei einem INPUT keinen Wert eingeben und direkt die [ENTER] -Taste drücken, so behält die entsprechende Variable ihren alten Wert.

Beispiele:

1Ø: INPUT A	Löscht das Anzeigenfeld und setzt ein Fragezeichen an den Anfang der 1. Zeile.
2Ø: INPUT "A="; A	Gibt "A=" aus und wartet auf eine Eingabe.
3Ø: INPUT "A=", A	Gibt "A=" aus, wartet auf eine Eingabe und zeigt diese dann am Anfang der 1. Zeile an. Der Prompt-String wird also gelöscht.
4Ø: INPUT "X=?"; X; "Y=?"; Y	Gibt erst das Prompt "X=?" aus, wartet auf eine Eingabe, löscht nach dem [ENTER] die Anzeige und gibt dann den Prompt "Y=?" aus.

INPUT

1. INPUT # **var liste**
2. INPUT # **"dateiname"; var liste**

var liste **ist variable** oder
variable, var liste

Abkürzung: I.#, IN.#, INP.#, INPU.#
Vergleiche: INPUT, PRINT #, READ

Wirkung:
Mit dem INPUT#-Befehl können Daten von einer Cassette geladen werden.

Anwendung:
Folgende Variable können im INPUT #-Befehl eingesetzt werden:

1. Standardvariable – A, B, C, A(7), D*, A(2Ø)* usw.
2. Einfache Variable – AA, B3, CP$ usw.
3. Feld-Variable – S(*), HP(*), K$(*) usw.

1) Übergabe von Daten an Standardvariablen

Um Daten von der Cassette den Standardvariablen zuweisen zu können, müssen die Namen der Variablen im Befehl angegeben werden.

```
INPUT# "DATA1"; A, B, X, Y
```

Der Befehl weist die Daten der Datei DATA1 den Variablen A, B, X und Y in der gegebenen Reihenfolge zu.

Um eine Folge von Standardvariablen und falls definiert, erweiterte Variable (A(27) und weiter) mit Daten von Cassette zu belegen, muß der ersten Variablen ein Stern (*) angehängt werden.

```
INPUT# "D2"; D*
```

Mit diesem Befehl werden die Daten der Cassette den Variablen D bis Z und A(27) und höher zugewiesen.

```
INPUT # A(1Ø)*  (ohne DIMensionierung)
```

Dieser Befehl überträgt die Daten der ersten gefundenen Datei von Cassette auf die Variablen A(1Ø) und höher (also J bis Z und A(27) und höher).

Achtung:

a) Wurde bereits mit einem DIM-Befehl ein Feld namens "A" definiert, kann keine erweiterte Variable der Form A( ) definiert werden.

b) Die Übertragung der Daten auf die Standardvariablen und Felder der Form A( ), d.h. A(27) und höher, wird solange fortgesetzt, wie Daten gefunden werden oder bis der Speicher des Rechners voll ist.

2) Datenübertragung auf einfache Variable

Werden im INPUT#-Befehl einfache Variablennamen eingegeben, so werden die Daten der Cassette auf diese Variablen übertragen.

```
INPUT# "DM!"; AB, Y1, XY$
```

Dieser Befehl überträgt die Daten von Cassette auf die Variablen AB, Y1 und XY$.

Achtung:

a) Numerische Daten müssen auf numerische Variable, String-Daten auf String-Variable übertragen werden. Andere Zuweisungen sind nicht möglich.

b) Im Programm-Datenspeicherbereich muß Raum für die einfachen Variablen frei gehalten werden, bevor der INPUT#-Befehl ausgeführt werden kann, sonst erhalten Sie eine Fehlermeldung. Die "Platzreservierung" kann mit einer Zuweisung ausgeführt werden.

```
AA=Ø [ENTER]
BI$="A" [ENTER]
INPUT# AA, BI$ [ENTER]
```

Für die Zuweisung können einfache numerische oder String-Werte benutzt werden.

3) Datenübertragung auf Feldvariable

Um Daten von der Cassette auf die Elemente eines Feldes zu übertragen, muß im Befehl der Name des Feldes in der Form feldname (*) angegeben werden.

```
5Ø: DIM B(5)
6Ø: INPUT# "DS4"; B(*)
```

Dieser Befehl überträgt die Daten der Datei DS4 auf die Variablen B(Ø) bis B(5) des Feldes B.

Achtung:

a) Numerische Daten müssen auf numerische Feldvariablen gleicher Länge, String-Daten müssen auf String-Feldvariable gleicher Länge übertragen werden. Wird dies nicht beachtet, erhalten Sie eine Fehlermeldung.

b) Sie müssen Raum im Programm-Datenspeicherbereich bereitstellen, bevor der INPUT#-Befehl ausgeführt wird, oder Sie erhalten eine Fehlermeldung. Benutzen Sie dazu den DIM-Befehl.

WICHTIG

Stimmt die Anzahl der Variablen im INPUT#-Befehl nicht mit der Zahl der Daten auf der Cassette überein, geschieht folgendes:

* Ist die Zahl der Daten in der Datei auf Cassette größer als die Zahl der Variablen des Befehls, so wird jeder Variablen ein Wert zugewiesen. Die überzähligen Daten werden ignoriert.

* Ist die Zahl der Daten in der Datei auf Cassette kleiner als die Zahl der Variablen des Befehls, so werden alle Daten der Datei den Variablen zugewiesen und die überzähligen Variablen behalten ihren alten Wert.
  Der Rechner wartet aber auf weitere Daten für diese Variablen. Unterbrechen Sie diesen Status mit der BRK -Taste.

* Wird der INPUT#-Befehl ohne Variable eingegeben, so erhalten Sie bei dem Versuch der Ausführung die Fehlermeldung ERROR 1.

LET

1. LET **variable = ausdruck**
2. **variable = ausdruck**

Abkürzung: LE.

Wirkung:
Der LET-Befehl weist Variablen Werte zu.

Anwendung:
Mit dem LET-Befehl weisen Sie der angegebenen Variablen den Wert des Ausdrucks zu. Sie können einer numerischen Variablen nur numerische Werte und einer String-Variablen nur String-Werte zuweisen. Um Werte aus der einen in die andere Form zu übertragen, muß eine der beiden Funktionen STR$ oder VAL eingesetzt werden.

Sie können das LET in einer Zuweisung weglassen, es sei denn, die Zuweisung folgt auf das THEN eines IF . . . THEN-Befehles. Dies ist der einzige Fall, wo das Wort LET im Befehl vorkommen muß.

Beispiele:

10: I=10	Weist I den Wert 10 zu.
20: A=5*I	Weist A den Wert 50 zu.
30: X$=STR$(A)	Weist X$ den Wert "50" zu.
40: IF I>=10 THEN LET Y$=X$+".00"	Weist Y$ den Wert "50.00" zu.

Für Drucker CE-126P

LPRINT

1. LPRINT **druckausdr**
2. LPRINT **druckausdr, druckausdr**
3. LPRINT **druckliste; druckausdr; . . . . ; druckausdr**

Abkürzung: LP., LPR., LPRI., LPRIN.

Vergleiche: PRINT, USING

Wirkung:

Mit dem LPRINT-Befehl werden Informationen auf der Option CE-126P ausgegeben.

Anwendung:

Wenn das serielle I/O-Interface durch das OPEN-Kommando aktiviert wurde, erfolgt die Ausgabe auf dem angeschlossenen Terminal.

* In der ersten Form des Befehls werden numerische Werte rechtsbündig und alphabetische Zeichen vom linken Rand des Papiers ausgedruckt. Wenn eine Zeile mehr als 24 Zeichen enthält, wird automatisch ein Zeilenvorschub-Befehl ausgeführt.

* In der Form 2 werden die 24 Zeichen einer Zeile in zwei Gruppen zu je 12 Zeichen aufgeteilt. Die Daten werden symmetrisch vom Komma ausgedruckt.

Werte, die mehr als 12 Zeichen lang sind, werden auf 12 Zeichen gekürzt. Numerische Werte werden dabei in Dezimalbruch und Zehnerpotenz ausgegeben. Strings werden nur bis zum 12. Zeichen ausgedruckt.

From 3 druckt die Werte links beginnend aus. Werden mehr als 24 Zeichen eingegeben, so wird der Druck in der nächsten Zeile fortgesetzt. Maximal können 96 Zeichen ausgedruckt werden.

* Der Drucker CE-126P hat Priorität, wenn sowohl der CE-126P und der CE-140P angeschlossen sind.

Beispiele:

```
10: A=10: B=20: X$="ABCD": Y$="XYZ"
20: LPRINT A
30: LPRINT X$
40: LPRINT A, B
50: LPRINT X$; A; B
60: LPRINT X$; Y$
70: LPRINT A * B
```

Für Drucker CE-140P

LPRINT

1. LPRINT **druckausdr**
2. LPRINT **druckausdr**, **druckausdr**, . . . , **druckausdr**
3. LPRINT **druckausdr**; **druckausdr**; . . . ; **druckausdr**
4. LPRINT . . . **druckausdr;**
   Bei dieser Form wird an das Ende von 1 und 3 oben ein ";" gesetzt.
5. LPRINT

Abkürzung: LP., LPR., LPRI., LPRIN.
Vergleiche: PRINT, CONSOLE, USING

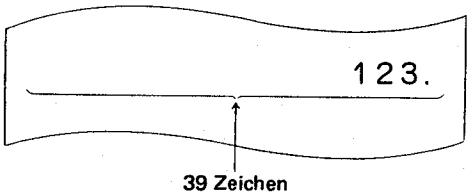
Wirkung:
Mit dem Befehl LPRINT werden Informationen auf den Drucker ausgegeben.

Anwendung:
Wenn mit dem OPEN-Kommando der serielle E/A-Anschluß geöffnet wurde, gibt das LPRINT-Kommando das Programm am seriellen E/A-Schnittstellen-Anschluß aus. (Siehe Seite 239.) Zur Rückkehr des Programm-Druck-Kommandos zum CE-140P das CLOSE-Kommando ausführen.

In Form 1 werden numerische Werte rechtsbündig und alphabetische Zeichen beginnend vom linken Papierrand innerhalb des mit dem CONSOLE-Kommando spezifizierten Bereiches (normalerweise 39 Zeichen) gedruckt. (Für Informationen zum CONSOLE-Kommando siehe Seite 232.)

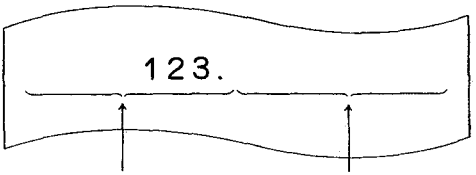
Beispiel: LPRINT 123 [ENTER] wird ausgeführt mit der Standard-Zeichengröße (Größe b)

1. Wenn 39 Zeichen pro Druckzeile mit dem CONSOLE-Kommando spezifiziert wurden



(Zeichen werden in diesem Bereich gedruckt.)

2. Wenn 20 Zeichen pro Druckzeile mit dem CONSOLE-Kommando spezifiziert wurden



In Form (2) ist die Anzahl Druckzeichen in Gruppen von 12 Zeichen unterteilt. Die spezifizierten Werte werden der Reihe nach ausgedruckt. Numerische Werte mit bis zu 12 Ziffern werden rechts und Zeichenstrings links auf dem Drucker ausgegeben. Wenn der auszudruckende Wert 12 Zeichen überschreitet, werden numerische Werte auf 12 Zeichen gerundet, während von Zeichenstrings die ersten 12 Zeichen (von links) gedruckt werden.

Die Maximale Anzahl Werte (Elemente, die spezifiziert und durch Kommata getrennt werden können) beträgt für Form (2) 8.

Werden mehr Elemente spezifiziert, tritt die Fehlermeldrung ERROR 1 auf.

In Form (3) werden die Werte beginnend vom linken Papierrand ausgedruckt. Wenn der auszudruckende Werte 24 Zeichen überschreitet, wird automatisch ein Zeilenvorschub ausgeführt. Bis zu maximal 96 Zeichen können gedruckt werden.

In Form (4) wird durch das Semikolon der Startpunkt des nächsten LPRINT-Befehls bestimmt. Dieser befindet sich direkt hinter dem zuletzt ausgegebenen Wert.

Hinweis: Wenn sich nur ein zu druckender Wert in einer durch einen LPRINT-Befehl zu druckenden Gleichung befindet, der auf einen in Form (4) geschriebenen Befehl folgt, diesen Wert mit ; "␣ " nach der Gleichung spezifizieren.

Beispiel:

```
10 A=50 : B=80
20 LPRINT "A * B=" ;
30 LPRINT A * B ; "␣ "
```

"␣ " verwenden, um Ergebnisse zu spezifizieren, die mit Zeile 30 nach dem Drucken in Zeile 20 gedruckt werden sollen.

Die Form (5) bewirkt kein Drucken, sondern nur einen Zeilenvorschub.

NEXT

1. NEXT **numerische variable**

Abkürzung: N., NE., NEX.
Vergleiche: FOR

Wirkung:
Der NEXT-Befehl kennzeichnet das Ende einer Gruppe von Befehlen, die in einer FOR/NEXT-Schleife wiederholt werden sollen.

Anwendung:
Der NEXT-Befehl tritt immer in Verbindung mit einem FOR-Befehl auf und wird deshalb dort ausführlich erklärt. Die Variable des NEXT muß mit der des dazugehörigen FOR übereinstimmen.

Beispiel:

```
10: FOR I=1 TO 10
20: PRINT I
30: NEXT I
```

ON . . . GOSUB

1. ON **ausdruck** GOSUB **ausdr liste**

ausdr liste **ist ausdruck**

oder

ausdruck, ausdr liste

Abkürzung: O. ; GOS., GOSU.
Vergleiche: GOSUB, GOTO, ON . . . GOTO

Wirkung:
Der ON . . . GOSUB-Befehl wird benutzt, um eine der gegebenen Unterroutinen abhängig von einem Wert auszuführen.

Anwendung:
Bei der Ausführung des ON . . . GOSUB-Befehls wird der **ausdruck** zwischen ON und GOSUB auf einen ganzzahligen Wert reduziert. Ist der Wert dieses Ausdrucks 1, so wird die erste der durch **ausdr liste** angegebenen Unterroutinen nach dem GOSUB ausgeführt. Ist der Wert 2, so wird die zweite ausgeführt usw. Nach dem RETURN aus der Unterroutine wird der Befehl, der direkt auf den ON ... GOSUB-Befehl folgt, ausgeführt.

Wenn der Wert des **Ausdrucks** kleiner 1 oder größer der Zahl der gegebenen Unterroutinen ist, wird das Programm mit dem auf den ON . . . GOSUB-Befehl folgenden Befehl fortgesetzt.

Achtung: In den Ausdrücken nach dem GOSUB dürfen keine Kommata vorkommen. Der Computer kann nicht Kommata in Ausdrücken von Kommata zwischen Ausdrücken unterscheiden.

Beispiel:

```
10: INPUT A
20: ON A GOSUB 100, 200, 300
30: END
100: PRINT "ERSTE"
110: RETURN
200: PRINT "ZWEITE"
210: RETURN
300: PRINT "DRITTE"
310: RETURN
```

Wird in Zeile 10 "1" eingegeben, so gibt der Rechner "ERSTE" aus. Auf "2" gibt er "ZWEITE" und auf "3" "DRITTE" aus. Auf andere Eingaben wird nichts ausgegeben.

ON . . . GOTO

1. ON **ausdruck** GOTO **ausdr liste**

ausdr liste **ausdruck**

oder

ausdruck, ausdr liste

Abkürzung: O.; G., GO., GOT.
Vergleiche: GOSUB, GOTO, ON . . . GOSUB

Wirkung:
Mit dem Befehl ON . . . GOTO wird es möglich, das Programm abhängig von einem Wert an einer bestimmten Stelle fortzusetzen.

Anwendung:
Bei der Ausführung des Befehls ON . . . GOTO wird der **ausdruck** zwischen ON und GOTO auf einen ganzzahligen Wert reduziert. Ist der Wert dieser Zahl 1, so wird das Programm an der ersten angegebenen Stelle fortgesetzt. Ist er 2, an der zweiten usw.

Ist der Wert des **Ausdrucks** kleiner 1 oder größer der Zahl der gegebenen Sprungziele, so wird der auf den ON . . . GOTO-Befehl folgende Befehl ausgeführt.

Achtung: In den Ausdrücken, die auf das GOTO folgen, dürfen keine Kommata vorkommen. Der Computer ist nicht in der Lage, Kommata, die in Ausdrücken auftreten, von Kommata, die zwischen Ausdrücken stehen, zu unterscheiden.

Beispiel:

```
10: INPUT A
20: ON A GOTO 100, 200, 300
30: GOTO 900
100: PRINT "ERSTE"
110: GOTO 900
200: PRINT "ZWEITE"
210: GOTO 900
300: PRINT "DRITTE"
900: END
```

Die Eingabe '1' läßt den Rechner 'ERSTE', '2' 'ZWEITE' und '3' 'DRITTE' ausgeben.

PAUSE

1. PAUSE **druck ausdr**
2. PAUSE **druck ausdr**, **druck ausdr**
3. PAUSE **druck liste**; **druck liste**; . . . ; **druck liste**

druck liste **druck ausdr**

oder

druck ausdr; **druck liste**

druck ausdr **ausdruck**

oder

USING format; ausdruck

Das **USING format** wird unter USING beschrieben.

Abkürzung: PAU., PAUS.
Vergleiche: LPRINT, PRINT, USING, WAIT

Wirkung:
Mit dem PAUSE-Befehl wird eine Ausgabe auf dem Display zeitlich begrenzt wiedergegeben.

Anwendung:
Der PAUSE-Befehl wird benutzt, um kurze Texte, Ergebnisse usw. anzuzeigen. Die Ausführung des Befehls ist die gleiche wie die des PRINT-Befehls. Mit einer Ausnahme: Nach der Ausgabe des Textes oder des Wertes macht der Computer eine Pause von ca. Ø,85 Sekunden, bevor der nächste Befehl ausgeführt wird. Ein WAIT-Intervall oder [ENTER] wird nicht gebraucht.

Die erste Form des PAUSE-Befehls gibt einen einzelnen Wert aus. Numerische Werte werden ganz rechts, String-Werte ganz links beginnend ausgegeben.

In Form 2 wird das Anzeigenfeld in Gruppen mit je 12 Spalten unterteilt. Die Werte werden vom ersten spezifizierten Wert an in aufsteigender Reihenfolge angegeben. Auch hier werden innerhalb des Blocks numerische Werte ganz links und String-Werte ganz rechts beginnend ausgegeben.

* Die Anzahl der Werte in Form 2 darf höchstens 2 betragen.
* Ist ein Wert mehr als 12 Spalten (Zeichen) lang, wird
  a) ein numerischer Wert als Dezimalbruch mit Exponent wiedergegeben,
  b) ein Stringwert auf seine ersten 12 Zeichen gekürzt.

Form 3 gibt die Werte in direkter Folge von der linken Seite des Bildschirms beginnend wieder.

Achtung: Wenn der eingegebene Wert von Form 3 das Maximum von 24 Spalten überschreitet, wird der überzählige Teil nicht angezeigt.

Beispiele:

10: A=10 : B=20 : X$="ABCDEF" : Y$="XYZ"

	ANZEIGE	
20: PAUSE A		10.
30: PAUSE X$	ABCDEF	
40: PAUSE X$, B	ABCDEF	20.
50: PAUSE Y$; X$	XYZABCDEF	
60: PAUSE A * B		200

PRINT

1. PRINT **druck ausdr**
2. PRINT **durck ausdr**, **durck ausdr**
3. PRINT **druck liste**
4. PRINT = LPRINT
5. PRINT = PRINT

druck liste **druck ausdr**

oder

druck ausdr; **druck liste**

druck ausdr **ausdruck**

oder

USING format; **ausdruck**

Das **USING format** wird bei USING beschrieben.

Abkürzung: P., PR., PRI., PRIN.
Vergleiche: LPRINT, USING, WAIT

Wirkung:
Mit dem PRINT-Befehl können Ausgaben auf der Anzeige des Computers oder einen Drucker gemacht werden.

Anwendung:
Mit dem PRINT-Befehl werden Text und Ergebnisse ausgegeben. Die erste Form des Befehls gibt einen einzelnen Wert aus. Numerische Werte werden ganz rechts, String-Werte ganz links beginnend wiedergegeben.

In diesen Fall werden die Werte (im Bereich der 12 Spalten) in einem Ausdruck von der rechten Seite des Displays her und die Zeichen von der linken Seite her angezeigt.

* Die Anzahl der Werte in Form 2 darf höchstens 2 betragen.
* Ist ein Wert länger als 12 Spalten (Zeichen), so wird
  a) ein numerischer Wert als Dezimalbruch mit Exponent wiedergegeben,
  b) ein String-Wert auf die ersten 12 Zeichen gekürzt ausgegeben.

In Form 3 des Befehls werden die Werte direkt hintereinander linksbündig angezeigt.

Beispiele:

10: A=123 : B=456 : X$="ABCDEF" : Y$="VWXYZ"

	ANZEIGE
20: PRINT X$, B	ABCDEF 456.
30: PRINT A; B	123.456.
40: PRINT X$; A	ABCDEF123.
50: PRINT Y$; B	VWXYZ456.

PRINT #

1. PRINT # **var liste**
2. PRINT # **"dateiname"; var liste**

var liste **variable**

oder

variable, var liste

Abkürzung: P.#, PR.#, PRI.#, PRIN.#
Vergleiche: INPUT#, PRINT, READ

Wirkung:
Mit dem PRINT#-Befehl werden Werte auf Cassette abgespeichert.

Anwendung:
Folgende Variablentypen können benutzt werden:

1. Standardvariablen – A, B, X, A(26), C*, A(1Ø)* usw.
2. Einfache Variable – AA, B2, XY$ usw.
3. Feld-Variable – B(*), CD(*), N$(*) usw.

1) Abspeichern von Standardvariablen

Um die Werte von Standardvariablen auf Cassette abzuspeichern, müssen die Namen der Variablen im PRINT#-Befehl angegeben werden.

```
PRINT# "DATEI1"; A, B, X, Y
```

Dieser Befehl speichert die Inhalte der Variablen A, B, X und Y in der Datei "DATEI1" auf Cassette ab.

Wenn Sie den Inhalt aller Variablen ab einer bestimmten abspeichern wollen, so geben Sie die Startvariable gefolgt von einem Stern (*) ein.

```
PRINT# "D2"; D*
```

Dieser Befehl speichert die Inhalte der Variablen D bis Z (und A(26) und höher, wenn definiert) in der Datei mit dem Dateinamen "D2" ab.

```
PRINT#E, X$, A(3Ø)*
```

Dieser Befehl speichert die Inhalte der Standardvariablen E und X$ sowie die Inhalte der erweiterten Variablen A(3Ø) und höher auf Cassette ab.

Achtung: A( ) kann nicht zur Definition nachfolgender Variablen benutzt werden, wenn A bereits durch einen DIM-Befehl definiert wurde. Ansonsten können die Variablen A(1) bis A(26) im PRINT#-Befehl genauso angegeben werden, wie die Variablen A bis Z bzw. A$ bis Z$.

2) Abspeicherung der Inhalte von einfachen Variablen

Die Inhalte von einfachen Variablen (Zwei-Zeichen-Variablen) werden abgespeichert, indem die Namen der Variablen im PRINT#-Befehl angegeben werden.

```
PRINT# "DM1"; AB, Y1, XY$
```

Dieser Befehl speichert die Inhalte der einfachen Variablen AB, Y1 und XY$ in der Datei mit dem Dateinamen "DM1" ab.

3) Abspeicherung der Inhalte von Feldvariablen

Die Inhalte aller Variablen eines Feldes können auf Cassette abgespeichert werden, indem der Name des Feldes im Befehl angegeben und die Variablen selbst mit einem Stern (*) umschrieben werden.

```
PRINT# "DS2"; X(*), Y(*)
```

Dieser Befehl speichert die Inhalte der Elemente der Felder X (X(Ø), X(1), . . .) und Y (Y(Ø), Y(1) . . .) in der Datei mit dem Dateinamen "DS2" ab.

Achtung: Es ist nicht möglich, den Inhalt eines oder einiger Elemente eines Feldes gesondert abzuspeichern.

* Wird der PRINT#-Befehl ohne Variable eingegeben, erhalten Sie die Fehlermeldung ERROR 1.

WICHTIG

Die Lokation von erweiterten (A(26) und höher), einfachen und/oder Feldvariablen muß vor Ausführung des PRINT#-Befehls im Programm/Datenspeicherbereich bereitgestellt werden. Sonst ist die Ausfuhrung des Befehls nicht möglich und Sie erhalten eine Fehlermeldung.

RADIAN

1. RADIAN

Abkürzung: RAD., RADI., RADIA.
Vergleiche: DEGREE, GRAD

Wirkung:
Mit dem RADIAN-Befehl wird der Winkel in der Einheit BOGENMASS interpretiert.

Anwendung:
Der Computer hat drei Möglichkeiten der Darstellung von Winkelwerten:

- ALT-GRAD
- BOGENMASS
- NEU-GRAD

Sie werden für die Argumente der Funktionen SINus, COSinus und TANgens sowie für die Ergebnisse der Umkehrfunktionen ArcusSiNus, ArcusCoSinus und ArcusTaNgens gebraucht.

Der RADIAN-Befehl wechselt die Darstellungsform auf Radialwert. Die Radialform gibt den Winkel als Bogenmaß A in Abhängigkeit vom Radius wieder. 360 Grad sind z. B. 2 * PI, da der Einheitskreis den Umfang 2 mal PI mal Radius hat.

Beispiel:

```
10: RADIAN
20: X=ASN 1          X hat den Wert 1.570796327 (=PI/2), Arcsin von 1.
30: PRINT X
```

RANDOM

1. RANDOM

Abkürzung: RA., RAN., RAND., RANDO.
Vergleiche: RND.

Wirkung:
Der RANDOM-Befehl setzt eine neue Startzahl für den Zufallszahlengenerator.

Anwendung:
Wenn mit der RND-Funktion Zufallszahlen erzeugt werden, so startet der Computer bei einer vorgegebenen Zahl. Der RANDOM-Befehl setzt die Startzahl auf einen neuen, zufällig gewählten Wert.

Die Startzahl für den Zufallszahlengenerator ist jedes Mal, wenn der Computer eingeschaltet wird, die gleiche. Auch die Sequenz der Zufallszahlen wiederholt sich damit, es sei dann, die Basis (oder Start-)zahl wird mit dem RANDOM-Befehl gewechselt. Diese Eigenschaft ist sehr wichtig, ist es doch möglich, den korrekten Ablauf eines Programms zu ermitteln, da auch dann, wenn im Programm RND-Befehle auftreten, das Verhalten des Rechners immer gleich sein muß. Mit dem RANDOM-Befehl verbessern Sie also das Zufallsverhalten Ihres Programms, da die Basiszahl des Zufallsgenerators auch zufällig ermittelt wird.

Beispiel:

```
10: RANDOM
20: X=RND 10
30: GOTO 20
```

Bei Start in Zeile 20 wird die Standardbasis des Zufallszahlengenerators eingesetzt. Bei Start in Zeile 10 wird eine neue Basis ermittelt.

READ

1. READ **var liste**

var liste **variable**

oder

variable, var liste

Abkürzung: REA.
Vergleiche: DATA, RESTORE

Wirkung:
Der READ-Befehl wird gebraucht, um Daten aus den DATA-Zeilen herauszulesen und Variablen zuzuweisen.

Anwendung:
Wenn einem Feld Basiswerte zugewiesen werden sollen, so ist es sinnvoll, diese Daten in DATA-Zeilen unterzubringen und sie von dort mit einem READ-Befehl in einer FOR/NEXT-Schleife in das Feld zu übertragen. Wird der erste READ-Befehl ausgeführt, so wird der erste Wert aus den DATA-Zeilen zugewiesen, mit dem zweiten READ-Befehl der zweite Wert usw. Die Variablen der READ-Anweisung und die Daten der DATA-Anweisung müssen in Bezug auf ihre Anzahl und ihren Typ zueinander passen, d. h. die Anzahl der Variablen muß gleich der Anzahl der Daten sein, einer numerischen Variablen kann nur ein numerischer Ausdruck und einer Textvariablen nur ein Textausdruck zugewiesen werden.

Wenn nötig, können dieselben Daten ein zweites Mal gelesen werden. Hierzu wird der RESTORE-Befehl gebraucht.

Beispiel:

```
10: DIM B(10)
20: WAIT 32
30: FOR I=1 TO 10
40: READ B(I)
50: PRINT B(I) * 2
60: NEXT I
70: DATA 10,20,30,40,50,60
80: DATA 70,80,90,100
90: END
```

Dimensionierung eines Feldes

Zuweisung der Werte aus den DATA-Zeilen in die Feldvariablen B(1) bis B(10); B(1) ist 10, B(2) ist 20 usw.

REM

1. REM **kommentar**

Wirkung:
Mit dem REM-Befehl können Kommentare in ein Programm eingefügt werden.

Anwendung:
Es ist sinnvoll, in ein Programm erläuternde Kommentare einzufügen. Es kann sich hierbei um Titel, Autorennamen, Daten der letzten Änderung, Anwendungshinweise usw. handeln.

Solche Kommentare können mit einem REM-Befehl eingefügt werden. Der REM-Befehl hat keinen Einfluß auf den Programmablauf. Er kann überall im Programm eingefügt werden.

Beispiel:

```
10: REM DIESE ZEILE HAT KEINEN EFFEKT
```

RESTORE

1. RESTORE
2. RESTORE **ausdruck**

Abkürzung: RES., REST., RESTO., RESTOR.
Vergleiche: DATA, READ

Wirkung:
Der RESTORE-Befehl ermöglicht es, eine DATA-Zeile mehrmals zu lesen oder die Reihenfolge, in der die DATA-Zeilen gelesen werden sollen, zu verändern.

Anwendung:
Normalerweise werden mit dem READ-Befehl die Daten aus den DATA-Zeilen eine nach dem anderen herausgelesen, in der Reihenfolge, in der sie stehen. Ist der letzte Wert aus einer DATA-Zeile herausgelesen, so wird, wenn vorhanden und abgefragt, der erste Wert aus der nächsten DATA-Zeile genommen.

Die erste Form des RESTORE-Befehls setzt den Zeiger an den Anfang der ersten DATA-Zeile zurück.

Die zweite Form setzt den Zeiger an den Anfang der DATA-Zeile, deren Zeilennummer gleich der im **ausdruck** angegebenen Zeilennummer ist.

Beispiel:

```
10: DIM B(10)
20: WAIT 32
30: FOR I=1 TO 10
40: RESTORE
50: READ B(I)
60: PRINT B(I) * I
70: NEXT I
80: DATA 20
90: END
```

Dimensionierung eines Feldes

Setzt den Zeiger an den Anfang der 1. DATA-Zeile. Den Elementen von B werden die Werte zugewiesen. Alle Elemente haben den Wert 20.

RETURN

1. RETURN

Abkürzung: RE., RET., RETU., RETUR.
Vergleiche: GOSUB, ON . . . GOSUB

Wirkung:
Mit dem RETURN-Befehl wird eine Unterroutine geschlossen, der Programmablauf wird mit dem Befehl, der auf den GOSUB-Befehl folgt, fortgesetzt.

Anwendung:
Eine Unterroutine kann mehr als ein RETURN enthalten. Erreicht das Programm einen dieser RETURN-Befehle, wird die Unterroutine geschlossen und das Programm mit dem Befehl hinter dem GOSUB bzw. ON . . . GOSUB, von dem der Rechner in die Unterroutine gesprungen ist, fortgesetzt. Wird ein RETURN ohne GOSUB ausgeführt, erhalten Sie die Fehlermeldung ERROR 5.

Beispiel:

```
10: GOSUB 100
20: END
100: PRINT "HALLO"
110: RETURN
```

Das Programm druckt "HALLO" aus und springt zurück in Zeile 20.

STOP

1. STOP

Abkürzung: S., ST., STO.
Vergleiche: END; CONT-Kommando

Wirkung:
Mit dem STOP-Befehl wird die Ausführung eines Programms gestoppt.

Anwendung:
Erreicht der Computer in einem Programm einen STOP-Befehl, so wird die Ausführung des Programms abgebrochen und Sie erhalten eine Meldung wie z.B. "BREAK IN 2Ø0", wenn das Programm in Zeile 2Ø0 gestoppt wurde. Sie können jetzt die Variablen des Programms überprüfen und so die Ausführung desselben kontrollieren. Die Ausführung des Programms kann mit dem CONT-Befehl fortgesetzt werden.

Beispiel:

```
1Ø: STOP
```

BREAK IN 1Ø erscheint auf der Anzeige.

TROFF

1. TROFF

Abkürzung: TROF.
Vergleiche: TRON

Wirkung:
Mit dem TROFF-Befehl wird der Trace-Modus ausgeschaltet.

Anwendung:
Die Ausführung des Befehls TROFF hat zu Folge, daß ein Programm wieder auf "normale" Weise ausgeführt wird.

Beispiel:

```
1Ø: TRON
2Ø: FOR I=1 TO 3
3Ø: NEXT I
4Ø: TROFF
```

Bei Ausführung des Programms werden die Zeilennummern ausgegeben:
1Ø, 2Ø, 3Ø, 3Ø und 4Ø

TRON

1. TRON

Abkürzung: TR., TRO.
Vergleiche: TROFF

Wirkung:
Der TRON-Befehl schaltet den Trace-Modus ein.

Anwendung:
Der Trace-Modus unterstützt die Fehlersuche in Programmen. Wird das Programm gestartet, wird die erste Zeile des Programms abgearbeitet und deren Zeilennummer angezeigt. Durch Drücken der [↓] -Taste wird die nächste Programmzeile abgearbeitet und die Zeilennummer angezeigt usw.
Das Programm der jeweils angezeigten Zeilennummer kann durch Drücken der [↑] -Taste angezeigt werden. Der Trace-Modus bleibt so lange eingeschaltet, bis er mit dem TROFF-Befehl wieder ausgeschaltet wird oder bis Sie die Tastenfolge [SHIFT] [C-CE] eingeben.

Beispiel:

```
10: TRON
20: FOR I=1 TO 3
30: NEXT I
40: TROFF
```

Bei Ausführung des Programms werden die Zeilennummern ausgegeben:
10, 20, 30, 30, 30 und 40

USING

1. USING
2. USING *"spezifikation"*

Abkürzung: U., US., USI., USIN.
Vergleiche: LPRINT, PAUSE, PRINT

Weitere Erklärung zu USING finden Sie im Anhang C.

Wirkung:
Mit dem USING-Befehl kann das Anzeigenfeld oder der Druck formatiert werden.

Anwendung:
Der USING-Befehl kann einzeln oder als Erweiterung der Befehle LPRINT, PAUSE oder PRINT eingesetzt werden. Der USING-Befehl erstellt eine Ausgabeformatierung, die für alle folgenden Ausgabefehle Gültigkeit hat, bis die Formatierung durch einen neuen USING-Befehl geändert wird.

Die Spezifikationen des Befehls bestehen aus einer in Anführungsstrichen (") gesetzten Zeichenkette. Die gültigen Zeichen dafür sind:

#	Anzahl der Stellen für einen numerischen Ausdruck
.	Dezimalpunkt
^	Wissenschaftliches Format
&	Anzahl der Stellen für einen Textausdruck

Die Spezifikation "####" formatiert die Ausgabe eines numerischen Ausdrucks mit 3 Ziffern, kein Dezimalpunkt und eine Vorzeichenstelle. Für numerische Ausdrücke muß immer eine Position für das Vorzeichen vorgesehen sein, auch, wenn nur positive Zahlen ausgegeben werden sollen.

Eine Formatspezifikation kann beide Formate enthalten, eins für numerische und eins für Textausdrücke.
Die Spezifikation "###.##&&&" formatiert eine Zahl mit zwei Vorkomma-, zwei Nachkommastellen und eine Stelle für das Vorzeichen sowie eine Zeichenkette mit 3 Zeichen.

Wird wie in Form 1 des Befehls, keine Spezifikation gegeben, so werden alle gesetzten Formate gelöscht und die Standardformatierung tritt wieder in Kraft.

Beispiele:

10: A=125 : X$="ABCDEF"

	ANZEIGE
20: PRINT USING "##.##^"; A	1.25E02
30: PRINT USING "&&&&&&&"; X$	ABCDEF
40: PRINT USING "####&&&"; A; X$	125ABC

(Siehe Anhang C für zusätzliche Informationen über die Verwendung von USING.)

WAIT

1. WAIT
2. WAIT **ausdruck**

Abkürzung: W., WA., WAI.
Vergleiche: PRINT

Wirkung:
Der WAIT-Befehl spezifiziert die Anzeigedauer einer Ausgabe.

Anwendung:
Bei normaler Ausführung eines Programms wartet der Computer nach einem PRINT-Befehl, bis die [ENTER] -Taste gedrückt wird. Der WAIT-Befehl weist den Computer nun an, eine Ausgabe für eine vorgegebene Zeit anzuzeigen und dann die Ausführung des Programms fortzusetzen, genau wie mit dem PAUSE-Befehl. Der **ausdruck** im WAIT-Befehl setzt die Länge des Intervalls fest. Er kann jeden Wert von Ø bis 65535 haben. Dabei bedeutet die Erhöhung um 1 eine Verlängerung des Intervalls um 1/59 Sekunden. WAIT Ø ist zu schnell, als daß die Ausgabe gelesen werden könnte; WAIT 65535 setzt das Intervall auf ca. 19 Minuten. WAIT ohne Ausdruck setzt den Computer auf seine normale Arbeitsweise zurück.

Beispiel:

```
1Ø: WAIT 59
```

Der Rechner wartet nach PRINT etwa 1 Sekunde.

Grafikbezogene Befehle

Die folgenden Befehle sind für die Grafikfunktionen des Farb-Matrixdruckers CE-140P bzw. Farb-Plotter/Druckers CE-515P. Der Ausdruck von Daten in grafischer Form kann so durchgeführt werden, wie in den Bedienungsanleitungen dieser Drucker beschrieben (d.h. Übermittlung von Formcodes per LPRINT-Befehl). Mit den folgenden Befehlen lassen sich grafische Ausdrucke jedoch wesentlich leichter gestalten.
Alle Werte der Druckerbefehle werden mit auf ganze Zahlen gerundeten Werten durchgeführt.

COLOR

1. COLOR **Ausdruck**

2. COLOR **Ausdruck, 7**

Abkürzung: COL., COLO.
Vergleiche: LLINE, RLINE

Wirkung:
Mit dem COLOR-Befehl wird die Farbe angewählt, mit denen Zeichen oder Linien im normalen oder erweiterten Farbmodus gedruckt werden.

Anwendung:
Mit Format 1 (COLOR Ausdruck) lassen sich vier verschiedene Farben durch Angabe eines Werts zwischen 0 und 3 bestimmen.

Mit Format 2 (COLOR Ausdruck, 7) lassen sich 8 verschiedene Farben durch Angabe eines Werts zwischen 0 und 7 bestimmen.

Für den Farb-Matrixdrucker CE-140P können beide Formate verwendet werden. Der Farb-Plotter/Drucker CE-515P akzeptiert jedoch nur Format 1.
Es können folgende Farben bestimmt werden:

CE-140P

Wert des Ausdrucks	0	1	2	3	4	5	6	7
Format 1	Schwarz	Violett	Grün	Rot				
Format 2	Schwarz	Violett	Rot	Magenta	Grün	Zyan	Gelb	Weiß

Achtung: Die mit dem Wert "7" zu bestimmende Farbe entspricht keiner auszudruckenden Farbe. Bei der Angabe dieses Werts erfolgt zwar eine Druckkopfbewegung mit Papiertransport, es wird jedoch nicht gedruckt.

CE-515P

Wert des Ausdrucks	0	1	2	3
Format 1	Schwarz	Violett	Grün	Rot

Wenn der Befehl COLOR in Format 2 gegeben wird, schaltet der Drucker automatisch auf erweiterten Farbmodus. In diesem Modus kann eine der oben aufgeführten acht Farben mit dem Befehl LLINE bzw. RLINE gewählt werden.
Der erweiterte Farbmodus wird beim Ausschalten des Computers aufgehoben.

Beispiel:

```
1Ø GRAPH
2Ø COLOR 1,7
```

Stellt den erweiterten Farbmodus ein und spezifiziert gleichzeitig "purpur" als die zu benutzende Farbe. (In diesem Fall muß CE-140P benutzt werden.)

CROTATE

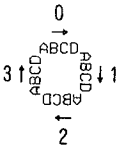
1. CROTATE **Ausdruck**

Abkürzung: CR., CRO., CROT., CROTA., CROTAT.

Wirkung:
Mit dem Befehl CROTATE wird die Druckrichtung und die Orientierung auszudruckender Zeichen bestimmt.

Anwendung:
Dieser Befehl ist nur wirksam, wenn der Drucker auf Grafikmodus geschaltet ist. Der Wert des Ausdrucks bestimmt hierbei die jeweilige Druckrichtung und Orientierung der Schriftzeichen.

* Wenn für den Ausdruck ein Wert zwischen 0 und 3 eingegeben wird, schaltet der Drucker nach dem folgenden Schema auf die zugehörige Druckrichtung und Orientierung um.



* Durch Geben des Befehls LTEXT schaltet der Drucker automatisch auf normale Druckrichtung und Orientierung der Zeichen, die dem Wert 0 entsprechen.

Beispiel 2:

```
 5 OPEN
10 GRAPH
20 GLCURSOR (200, -30)
30 FOR Z=0 TO 3
40 CROTATE Z
50 LPRINT "PABCD"
60 NEXT Z
70 LTEXT
80 LPRINT
90 END
```

Dieser Befehl muß für den CE-515P gegeben werden.

CSIZE

1. CSIZE **Ausdruck**

Abkürzung: CSI., CSIZ.

Wirkung:
Mit dem Befehl CSIZE wird die Größe der auszudruckenden Zeichen festgelegt.

Anwendung:
Die Größe der auszudruckenden Zeichen kann bestimmt werden, indem man für den Ausdruck einen Wert eingibt, der innerhalb des folgenden Bereichs liegt.

CE-140P	CE-515P
1 bis 63	1 bis 15

Hierbei entspricht 1 der kleinsten Zeichengröße, so daß bei Eingabe von 2, 3, 4 .... eine doppelte, dreifache, vierfache ..... Zeichengröße mit entsprechendem Zeichen- und Zeilenabstand erhalten wird.

Beispiel:
CSIZE 1

Zeichengröße : 0,8 mm (B) × 1,2 mm (H) (4 × 6 Schritte)
Zeichenabstand: 1,2 mm (6 Schritte)
Zeilenabstand : 2,4 mm (12 Schritte)

Achtung:
Nach dem Einschalten entspricht der Betriebszustand des Druckers CSIZE 2.

GLCURSOR

1.GLCURSOR **(Ausdruck 1, Ausdruck 2)**

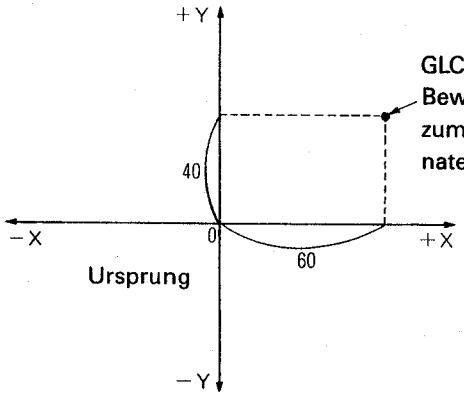
Abkürzung: GL., GLC., GLCU., GLCUR., GLCURS., GLCURSO.

Wirkung:
Mit dem Befehl GLCURSOR können Bewegungen des Druckkopfes ausgeführt werden.

Anwendung:
Dieser Befehl ist nur im Grafikmodus zur Bewegung des Druckkopfes in X- und Y-Achsenrichtung vom Ursprung aus wirksam.

Der Druckkopf bewegt sich hierbei zu dem Punkt des Koordinatenpaars, das mit Ausdruck 1 (X-Koordinate) und Ausdruck 2 (Y-Achse) bestimmt wurde.

Die Werte für die beiden Ausdrücke müssen innerhalb des Bereichs von −999 bis 999 liegen. Der kleinste Bewegungsschritt ist 0,2 mm in beide Richtungen.



Achtung:
Zur Bewegung des Druckkopfes in Y-Richtung muß Papiertransport rückwärts erfolgen.
Näheres zum Ausschnitt-Bereich (Scissoring) ist der Bedienungsanleitung des Drukkers zu entnehmen.

Beispiel:

```
10 GRAPH
20 GLCURSOR (60, 40)   ← Bewegt den Stift zum Punkt der
                         Koordinaten (X = 60, Y = 40)
```

GRAPH

1. GRAPH

Abkürzung: GRAP.
Vergleiche: LTEXT

Wirkung:
Mit dem Befehl GRAPH wird der Drucker auf Grafikmodus umgeschaltet.

Anwendung:
Bei Ausführung dieses Befehls schaltet der Drucker zum Zeichnen einer grafischen Darstellung von Textmodus auf Grafikmodus um.

* Durch Ausführung des Befehls LLIST bzw. nach manuellem Ausdruck mit dem angeschlossenen CE-140P wird der Grafikmodus aufgehoben, und der Drucker schaltet auf Textmodus zurück.

* Zum Ausdrucken von Zeichen im Grafikmodus kann der Befehl LPRINT in einem der beiden folgenden Formate eingesetzt werden:
(1) LPRINT "P Zeichenfolge"
(2) LPRINT "P" { + ; } Zeichenfolge-Variable

Achtung:
1. Zur Unterbrechung des Druckvorgangs durch Drücken der [BRK]-Taste beim Zeichnen einer grafischen Darstellung muß folgendes eingegeben werden:
LPRINT " " [ENTER]
Wenn diese Eingabe versäumt wird, werden nachfolgend gegebene Befehle unter Umständen nicht ordnungsgemäß ausgeführt.

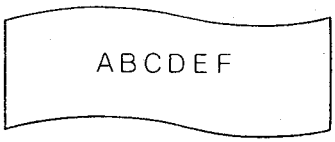
2. Zur Rückführung des Druckkopfes zur linken Ausgangsstellung wird folgendes eingegeben:
LTEXT [ENTER]
LPRINT " " [ENTER]
In diesem Fall wird der Grafikmodus jedoch aufgehoben.

3. Wird einer der nur im Grafikmodus wirksamen Druckerbefehle (CROTATE, GLCURSOR usw.) im Textmodus verwendet, so wird nur das Anweisungszeichen ausgedruckt.

Beispiel:

```
 5 OPEN
1Ø GRAPH
2Ø GLCURSOR (2ØØ, −3Ø)
3Ø LPRINT "PABCDEF"
4Ø LTEXT
5Ø LPRINT
6Ø END
```

Dieser Befehl muß für den CE-515P gegeben werden.



LF

1. LF
2. LF **Ausdruck**

Wirkung:
Dem Befehl LF bewirkt Zeilenvorschub.

Anwendung:
Dieser Befehl ist nur im Textmodus wirksam.
* Mit Format 1 erfolgt Zeilenvorschub um eine Zeile.
* Mit Format 2 wird das Papier der angegebenen Zeilenzahl entsprechend transportiert. Der für den Ausdruck angegebene Wert muß zwischen -999 und 999 liegen.
* Bei Angabe eines positiven Werts erfolgt Zeilenschritt vorwärts, bei Angabe eines negativen Werts Zeilenschritt rückwärts.

Achtung:
Zeilenvorschub durch den Befehl LF erfolgt in dem Zeilenabstand, der mit dem Befehl CSIZE bestimmt wurde.

Beispiel:

```
LF1Ø          ← Papiervorsehub um 10 Zeilen.
```

LLINE

1. LLINE **[(Ausdruck 1, Ausdruck 2)]–(Ausdruck 3, Ausdruck 4) [,Ausdruck 5][,Ausdruck 6][,B]**

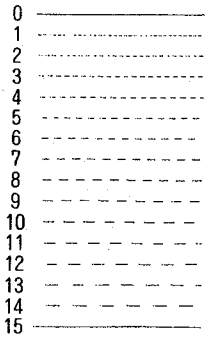
Abkürzung: LLIN.
Vergleiche: RLINE, COLOR

Wirkung:
Der Befehl LLINE verbindet zwei gegebene Punkte durch eine Linie.

Anwendung:
Dieser Befehl ist nur im Grafikmodus wirksam und wird verwendet, um den mit den mit (Ausdruck 1, Ausdruck 2) bestimmten Koordinatenpunkt durch eine Linie mit einem zweiten Koordinatenpunkt (Ausdruck 3, Ausdruck 4) zu verbinden.

* Die Angaben für (Ausdruck 1, Ausdruck 2) können ausgelassen werden, wenn als Ausgangspunkt für die Linie die momentane Position des Druckkopfes angenommen werden soll.

* Durch eine entsprechende Angabe für Ausdruck 5 (Wert zwischen 0 und 15) kann eine der folgenden Arten von Linien bestimmt werden. Der definierte Wert für Ausdruck 5 ist 0.



* Ausdruck 6 ist für die Bestimmung der Farbe, in der die Linie gezogen wird. Mit dem erweiterten Farbmodus kann der Wert für Ausdruck 6 innerhalb des Bereichs von 0 bis 7 liegen, mit dem normalen Farbmodus von 0 bis 3. (Die Farben mit zugehörigen Werten werden unter dem Befehl COLOR aufgeführt.)

* Ausdruck 6 kann ausgelassen werden. In diesem Fall werden zuvor gegebene Werte angenommen.

Beispiel:

```
 5 OPEN
1Ø GRAPH : RANDOM

2Ø GLCURSOR (24Ø, −12Ø)
25 SORGN
3Ø FOR J=Ø TO 34Ø STEP 2Ø
4Ø A=1Ø7*COS J
5Ø B=1Ø7*SIN J

6Ø R=RND 4−1

7Ø LLINE (Ø, Ø)−(A, B), Ø, R
8Ø NEXT J

9Ø LTEXT
1ØØ LPRINT


11Ø END
```

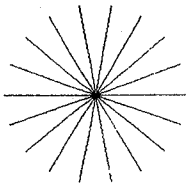
← Dieser Befehl muß für den CE-515P gegeben werden.

← Bewegt den Druckkopf zur ungefähren Mitte des Papiers, die als Koordinatenursprung zum Ausdruck einer grafischen Darstellung angenommen wird.

← R wird eine Zufallszahl zwischen 0 und 3 zugeordnet.

← Die Farbe wird von dem jeweiligen Wert für R bestimmt.

Der Drucker schaltet auf Textmodus zurück, wobei der Druckkopf zu seiner Ausgangsposition (ganz links) zurückläuft.



Die Darstellung wird in Farbe ausgedruckt.

* Wenn B am Ende des Befehls LLINE gesetzt wird, erhält man ein Rechteck, wobei der durch (Ausdruck 1, Ausdruck 2) definierte Koordinatenpunkt sowie der durch (Ausdruck 3, Ausdruck 4) definierte Koordinatenpunkt als sich gegenüberliegende Eckpunkte des Rechtecks (Endpunkte der Diagonalen) angenommen werden.

Beispiel:
LLINE (1Ø, 2Ø) − (2ØØ, −2Ø), 2, Ø, B

Achtung:
(Ausdruck 1, Ausdruck 2) darf zur Darstellung von Rechtecken nicht ausgelassen werden.

Befehlsanweisung zur Darstellung miteinander verbundener Linien

Durch Verwendung des folgenden Formats für LLINE werden fortlaufend gezeichnete, miteinander verbundene Linien erhalten.

LLINE **(Ausdruck 1, Ausdruck 2) – (Ausdruck 3, Ausdruck 4) – (Ausdruck 5, Ausdruck 6) – (Ausdruck 7, Ausdruck 8) – (Ausdruck 9, Ausdruck 10) – (Ausdruck 11, Ausdruck 12), Ausdruck, Ausdruck**

Mit diesem Befehlsformat erhält man eine Linie von dem durch (Ausdruck 1, Ausdruck 2) definierten Punkt zu dem durch (Ausdruck 3, Ausdruck 4) definierten Punkt, eine Linie von dem durch (Ausdruck 3, Ausdruck 4) definierten Punkt zu dem durch (Ausdruck 5, Ausdruck 6) definierten Punkt usw, wobei die Linien durchgehend gezeichnet werden. Es können bis zu sechs Paare von Ausdrücken angegeben werden.

LTEXT

1. LTEXT

Abkürzung: LT., LTE., LTEX.
Vergleiche: GRAPH

Wirkung:
Der LTEXT-Befehl dient zum Einstellen des Text-Betriebs.

Anwendung:
Dieser Befehl dient zum Einschalten des Text-Betriebs des Druckers, zum Ausdrucken von Buchstaben und Zahlen.

Achtung:
Der Drucker wird nach Ausführung des LLIST-Befehls oder nach manuellem Drucken mit dem CE-140P automatisch auf Text-Betrieb gestellt.
Den Operationsmodus stets mit dem GRAPH- oder dem LTEXT-Befehl eingeben, wenn die [BRK] -Taste nach dem automatischem Ausschalten des Netzstroms zum Computer gedrückt wird oder wenn die Stromversorgung von Computer oder Drukker wieder eingeschaltet wird.

Beispiel:
```
LTEXT          ← Schartet auf Textmodus.
```

RLINE

1 RLINE **(Ausdruck 1, Ausdruck 2) – (Ausdruck 3, Ausdruck 4)[, Ausdruck 5] [Ausdruck 6][,B]**

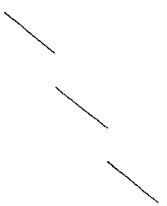
Abkürzung: RL., RLI., RLIN.
Vergleiche: LLINE, COLOR

Wirkung:
Der RLINE-Befehl dient zum Ziehen einer Linie zwischen den beiden Punkten, festgelegt durch die relativen Koordinaten.

Anwendung:
Dieser Befehl ist nur in der Grafik-Betriebsart wirksam.
Der RLINE-Befehl unterscheidet sich insofern von dem LLINE-Befehl, daß der LLINE-Befehl die Herkunft von Koordinaten festgelegt durch den SORGN-Befehl, als Bezugskoordinaten nimmt und die Stelle jedes Punktes durch Koordinate von dieser Herkunft bestimmt, während der RLINE-Befehl die derzeitige Position des Stiftes als die Herkunft der Koordinaten annimmt und die Stelle des nächsten Punktes durch Koordinate von dieser Herkunft bestimmt.

Beispiel:

```
 5 OPEN
10 GRAPH
20 FOR A=1 TO 3
30 RLINE (0, -40)-(60, -50)
40 NEXT A
50 LTEXT
60 LPRINT
70 END
```



* Die Beschreibung zu den Befehlen ist identisch mit der zum LLINE-Befehl, mit der Ausnahme, daß (Ausdruck 1, Ausdruck 2) nicht weggelassen werden können.

SORGN

1. SORGN

Abkürzung: SO., SOR., SORG.

Wirkung:
Der SORGN-Befehl dient zum Ändern der Koordinaten zum Aufzeichnen mit dem Stift.

Anwendung:
Dieser Befehl ist nur in der Grafik-Betriebsart wirksam und dient zur Bestimmung der derzeitigen Position des Stiftes als neue Herkunft von Koordinaten.
Das Aufzeichnen einer Grafik ist für den Drucker mitunter schwierig, wenn sich die Herkunft der Koordinaten am linken Ende des Papiers befinden. In diesem Fall, den Stift durch Ablauf des GLCURSOR-Befehls in eine arbiträre Position auf dem Papier führen und anschließend diese Position durch den SORGN-Befehl als die Herkunft des Koordinaten festlegen. Dies ermöglicht das Aufzeichnen einer Grafik mit der derzeitigen Stiftposition als Bezugspunkt.

Beispiel:

```
1Ø GRAPH
2Ø GLCURSOR (6Ø, 4Ø)
3Ø SORGN
```

← Spezifiziert die derzeitige Position (X = 60, Y = 40) des Stiftes als neuen Ausgang der Koordinaten)

10.3 FUNKTIONEN

VERTEILUNGSFUNKTIONEN

Verteilungsfunktionen sind eine Funktionsgruppe, die einen oder mehr numerische Werte als Argument haben und einen numerischen Wert liefern. Sie umfassen Funktionen zur Berechnung von Wahrscheinlichkeiten und Punkte für die verschiedenen Verteilungsarten.

BDS

1. BDS(p,n,r) (Binominalverteilung)

Abkürzung: BD.

BDS ist eine Verteilungsfunktion, die die Wahrscheinlichkeit für x Vorkommnisse bei n Versuchen für ein Ereignis mit der Wahrscheinlichkeit p in der Binominalverteilung liefert.

Beispiel: BDS(0.25,7,3)⟶Ø.173Ø35

Wertebereich: n und r müssen natürliche Zahlen (positive ganze Zahlen) kleiner als 10^{10} oder gleich 0 sein.

$\frac{n!}{(n-r)!}$ und $\frac{n!}{r!}$ müssen kleiner als 10^{100} sein.

Zum Beispiel bei N=10000 gilt $r \leqq 33$ oder $r \geqq 967$.

FDS

1. FDS(n_1,n_2,x) (F-Verteilung)

Abkürzung: FD.

FDS ist eine Verteilungsfunktion, die das obere Wahrscheinlichkeitsintegral p für ein gegebenes x für die F-Verteilung liefert.

Beispiel: FDS(2,3,5.5)⟶Ø.Ø992

Wertebereich: n_1 und n_2 müssen natürliche Zahlen kleiner als 10^{10} sein. Wenn für x eine negative Zahl spezifiziert wird, wird das Ergebnis gleich 1.

NDS

1. NDS(x) (Normalverteilung)

Abkürzung: ND.

NDS ist eine Verteilungsfunktion, die das obere Wahrscheinlichkeitsintegral p der Normalverteilung für ein gegebenes x liefert.

Beispiel: NDS(0.95)——→0.1711

Wertebereich: Diese Funktion liefert einen Wert mit einer Genauigkeit von bis zu 4 Dezimalstellen.

PDS

1. PDS(m,x) (Poisson-Verteilung)

Abkürzung: PD.

PDS ist eine Verteilungsfunktion, die die Wahrscheinlichkeit von x Vorkommnissen eines Ereignisses in der Poisson-Verteilung mit Mittelwert m liefert.

Beispiel: PDS(1.8,0)——→0.165299

Wertebereich: x muß eine ganze Zahl zwischen 0 und 69 sein. m muß bei $x \geqq 1$ im Bereich $0 \leqq m < 10^{100/x}$ liegen, und für $x=0$ muß $m>0$ gelten.

Hinweis: Die Grundgesamtheit muß ausreichend groß sein.

PFD

1. PFD(n_1,n_2,p) (Punkt der F-Verteilung)

Abkürzung: PF.

PFD ist eine Verteilungsfunktion, die den Punkt x für ein gegebenes oberes Wahrscheinlichkeitsintegral p der F-Verteilung mit den Freiheitsgraden n_1 und n_2 liefert.

Beispiel: PFD(2,3,0.01)⟶30.82

Wertebereich: n_1 und n_2 müssen natürliche Zahlen kleiner als 10^{10} sein.
p muß als numerischer Wert im Bereich 0.001 bis 1.000 mit Inkrementen von 0.001 spezifiziert werden.
Wenn jedoch n_1 oder n_2 kleiner gleich 39 ist, muß das jeweils andere n kleiner gleich 999 sein.
Bei $n_1 = 999$ und $n_2 = 39$ werden etwa 21 Minuten für die Berechnung benötigt.

PND

1. PND(p) (Punkt der Normalverteilung)

Abkürzung: PN.

PND ist eine Verteilungsfunktion, die den Punkt x für ein gegebenes oberes Wahrscheinlichkeitsintegral p in der Normalverteilung liefert.

Beispiel: PND(0.05)⟶1.645

Wertebereich: p muß als numerischer Wert im Bereich 0.001 bis 0.999 mit Inkrementen von 0.001 spezifiziert werden.
Das Ergebnis weist 4 signifikante Stellen auf.

PXD

1 PXD(n,p) (Punkt der χ^2-Verteilung)

Abkürzung: PX.

PXD ist eine Verteilungsfunktion, die den Punkt x für ein gegebenes oberes Wahrscheinlichkeitsintegral p der χ^2-Verteilung mit Freiheitsgrad n liefert.

Beispiel: PXD(4,0.01)⟶13.28

Wertebereich: n muß eine natürliche Zahl kleiner als 10^{10} sein.
p muß als numerischer Wert im Bereich 0.001 bis 1.000 mit Inkrementen von 0.001 spezifiziert werden.

PTD

1. PTD(n,p) (Punkt der t-Verteilung)

Abkürzung: PT.

PTD ist eine Verteilungsfunktion, die den Punkt x für ein gegebenes oberes Wahrscheinlichkeitsintegral p der t-Verteilung mit Freiheitsgrad n liefert.

Beispiel: PTD(4,0.01)⟶3.747

Wertebereich: n muß eine natürliche Zahl kleiner als 10^{10} sein.
p muß als numerischer Wert im Bereich 0.001 bis 0.999 mit Inkrementen von 0.001 spezifiziert werden.

TDS

1. TDS(n,x) (t-Verteilung)

Abkürzung: TD.

PTD ist eine Verteilungsfunktion, die das obere Wahrscheinlichkeitsintegral p für ein gegebenes x der t-Verteilung mit Freiheitsgrad n liefert.

Beispiel: TDS(4,1.533)⟶0.1

Wertebereich: n muß eine natürliche Zahl kleiner als 10^{10} sein.

XDS

1. XDS(n,x) (χ^2-Verteilung)

Abkürzung: X.

PTD ist eine Verteilungsfunktion, die das obere Wahrscheinlichkeitsintegral p für ein gegebenes x der x^2-Verteilung mit Freiheitsgrad n liefert.

Beispiel: XDS(4,7.8)⟶0.0992

Wertebereich: n muß eine natürliche Zahl kleiner als 10^{10} sein.
Wenn für x eine negative Zahl spezifiziert wird, wird das Ergebnis gleich 1.

PSEUDOVARIABLE

Pseudovariable sind eine Gruppe von Funktionen, die ohne Argument wie einfache Variable behandelt werden.

INKEY$

1. INKEY$

INKEY$ ist eine String-Pseudovariable, die den Wert der letzten, auf der Tastatur gedrückten Taste zugewisen bekommt. Die Tsten [ENTER], [C·CE], [SHIFT], [DEF], [SML], [↑]-Pfeil, [↓]-Pfeil, [►], [►]-Pfeil, [STAT], [BASIC], und wissenschaflliche Funktionstasten haben den Wert Nul (= leer). Mit INKEY$ können Eingaben über die Tastatur ohne ein abschließendes [ENTER] gemacht werden.

```
 5: WAIT 50
10: A$= INKEY$
20: B=ASC A$
30: IF B=0 THEN GOTO 10
40: IF B= . . .
```

Die Zeilen ab Zeile 40 können eine Tastatur-Testroutine enthalten (z. B. 40 PRINT A$). Wenn das Programm gestartet wird, ist der Wert von INKEY$ = NUL, da die letzte gedrückte Taste [ENTER] war.

- Wenn sich am Anfang des Programms ein INKEY$ Befehl befindet, kann es vorkommen, daß beim Starten des Programms die Starttaste vom INKEY$ Befehl gelesen wird. Beispielsweise kann es im folgenden Programm
  10 "Z" : Z$ = INKEY&
  vorkommen, daß die Taste [Z] gelesen wird, wenn das Programm durch Drücken der Tasten [DEF] [Z] gestartet wird.

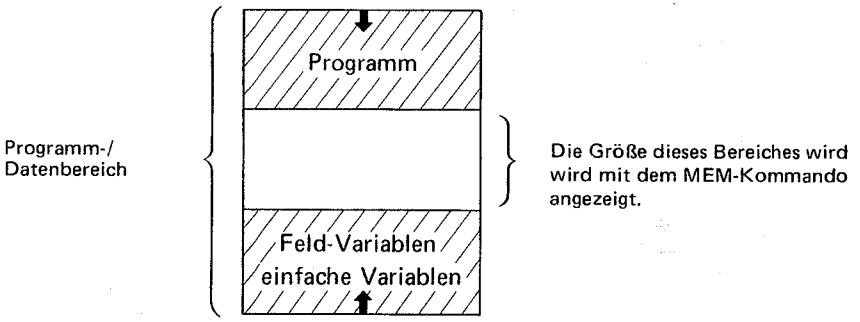
MEM

1. MEM

Abkürzung: M.

Wirkung:
Ermittelt die Anzahl der freien Bytes im Programmspeicher.

Anwendung:
Liefert die Anzahl der freien Bytes (der nicht durch ein Programm, Feldvariable oder einfache Variable belegte Bereich) im Programm/Daten-Bereich.



Die Programmgröße (in Bytes) kann durch folgende Eingabe angezeigt werden:

Beispiel: Computer mit RAM-Karte CE-212M

RUN-Modus

CLEAR [ENTER]	(löscht die einfachen Variablen, die Feld-Variablen usw.)
6878-MEM [ENTER]	(Es wird die Programmgröße – in Bytes – angezeigt.)

↑
└── Eingabe der gesamten RAM-Speichergröße (in Bytes).

RAM-Speichergröße (in Bytes)

Die Program- und Datenkapazitäten der einzelnen RAM-Karten sind auf Seite 125 aufgeführt.

PI

1. PI

PI ist eine numerische Pseudovariable, die den Wert π hat. Wie alle anderen Zahlen wird auch der Wert von PI mit einer Genauigkeit von 10 Stellen wiedergegeben (3.141592654).

NUMERISCHE FUNKTIONEN

Numerische Funktionen sind eine Gruppe von mathematischen Funktionen, die aus einem numerischen Argument ein numerisches Ergebnis errechnen. Die Gruppe enthält trigonometrische und logarithmische Funktionen, sowie solche, die ganze Zahlen und Beträge errechnen bzw. Vorzeichen ermitteln. Viele BASIC-Dialekte fordern, daß das Argument der Funktion in Klammern steht. Dieser Computer braucht Klammern nur dann, wenn in einer komplexen Berechnung das Argument der Funktion genau abgegrenzt werden muß.

LOG 10Ø+10Ø wird als

(LOG 10Ø) + 10Ø und nicht als LOG (10Ø + 10Ø)

interpretiert.

ABS

1. ABS **numerischer ausdruck**

Errechnet den Absolut-Betrag des angegebenen Arguments.

ABS −1Ø ist gleich 1Ø

ACS

1. ACS **numerischer ausdruck**

Errrechnet den Arcuscosinus des Arguments. Arcuscosinus ist die Umkehrfunktion des Cosinus. Das Ergebnis ist also der Winkel zu dem angegebenen Wert. Die Darstellung des Ergebnisses ist abhängig vom Winkelmodus des Computers (DEG, RAD, GRAD).

ACS .5 ist im DEG-Modus gleiche 6Ø

ASN

1. ASN **numerischer ausdruck**

Errechnet den Arcussinus des angegebenen Argumens. Arcussinus ist die Umkehrfunktion des Sinus. Die Darstellung des Ergebnisses ist abhängig vom Winkelmodus des Computers (DEG, RAD, GRAD).

ASN .5 ist im DEG-Modus gleich 3Ø

ATN

1. ATN **numerischer ausdruck**

Das Ergebnis ist der Arcustangens des Arguments. Arcustangens ist die Umkehrfunktion des Tangens. Die Darstellung des Ergebnisses ist abhängig vom Winkelmodus des Computers (DEG, RAD, GRAD).

ATN 1 ist im DEG-Modus gleich 45.

COS

1. COS **numerischer ausdruck**

Errechnet den Cosinus eines Winkels. Die Darstellung des Ergebnisses ist abhängig vom Winkelmodus des Computers (DEG, RAD, GRAD).

COS 6Ø ist im DEG-Modus gleich .5

CUR

1. CUR **numerischer ausdruck**

Errechnet die Kubikwurzel des gegebenen Arguments.

CUR8 ist gleich 2.

EXP

1. EXP **numerischer ausdruck**

Errechnet die Potenzen der Zahl e (= 2.718181828, die Basis des Logarithmus naturalis).

EXP 1 ist gleich 2.718291829

FACT

1. FACT **numerischer ausdruck**

Errechnet die Fakultät des gegebenen Arguments.

FACT5 ist gleich 120.

INT

1. INT **numerischer ausdruck**

Das Ergebnis ist der ganzzahlige Anteil des Arguments.

INT PI ist gleich 3.

LN

1. LN **numerischer ausdruck**

Errechnet den Logarithmus zur Basis e (2.718281828) des Arguments.

LN 1ØØ ist gleich 4.6Ø517Ø186

LOG

1. LOG **numerischer ausdruck**

Errechnet den Logarithmus zur Basis 1Ø des Arguments.

LOG 1ØØ ist gleich 2.

NCR

1. NCR **(numerischer Auschruck, numerischer Ausdruck)**

Berechnet den Wert der Kombinationen.

Die Auzahl der Kombinationen von r Objekten aus n verschiedenen Objekten kann als nCr=nPr/r! ausgedrückt werden.

NPR

1. NPR **(numerischer Ausdruck, numerischer Ausdruck)**

Berechnet den Wert der Permutationen.

Die Aanzahl der Permutationen vor r Objekten aus n verschiedenen Objekten kann als nPr=n!/(n−)! ausgedrückt werden.

RCP

1. RCP **numerischer ausdruck**

Errechnet den Reziprokwert des gegebenen Arguments.

RCP5 ist gleich 0.2.

RND

1. RND **numerischer ausdruck**

Erzeugt Zufallszahlen. Ist das Argument kleiner 1, aber größer Ø, so ist das Ergebnis kleiner 1 und größer oder gleich Ø. Ist das Argument eine ganze Zahl und größer oder gleich 1, so ist das Ergebnis kleiner oder gleich dem Argument und größer oder gleich 1. Ist das Argument keine ganze Zahl und größer 1, so ist das Ergebnis kleiner oder gleich der nächstgrößeren ganzen Zahl des Arguments und größer oder gleich 1.

Argument	Ergebnis	
	Untergrenze	Obergrenze
.5	Ø<	<1
2	1	2
2.5	1	3

Die Sequenz ist jedes Mal die gleiche, wenn der Computer eingeschaltet wird, da die Basiszahl immer die gleiche ist. Um diese Basis-Zahl zufällig zu ändern, brauchen Sie den RANDOM-Befehl.

ROT

1. **numerischer ausdruck** ROT **numerischer ausdruck**

Errechnet die Wurzel des angegebenen Arguments.

125ROT3 ist gleich 5.

(Beispiel: $\sqrt[3]{125}$ muß als 125 ROT 3 eingegeben werden.)

SGN

1. SGN **numerischer ausdruck**

Das Ergebnis ist abhängig vom Vorzeichen des Arguments. Ist das Argument positiv, ist das Ergebnis gleich 1; ist das Argument negativ, so ist das Ergebnis gleich −1; wenn das Argument gleich Ø ist, ist auch das Ergebnis gleich Ø.

SGN −5 ist gleich −1.

SIN

1. SIN **numerischer ausdruck**

Errechnet den Sinus des Winkelarguments. Die Darstellung ist abhängig von dem Winkelmodus des Computers (DEG, RAD, GRAD).

SIN 3Ø ist im DEG-Modus gleich .5.

SQR

1. SQR **numerischer ausdruck**

Errechnet die Quadratwurzel des Arguments. Das Ergebnis ist identisch mit der Anwendung der speziellen [√‾] -Taste der Tastatur.

SQR 4 ist gleich 2.

SQU

1. SQU **numerischer ausdruck**

Errechnet das Quadrat des gegebenen Arguments.

SQU3 ist gleich 9.

TAN

1. TAN **numerischer ausdruck**

Errechnet den Tangens des Winkelarguments. Die Darstellung ist abhängig vom Winkelmodus des Computer (DEG, RAD, GRAD).

TAN 45 ist im DEG-Modus gleich 1.

TEN

1. TEN **numerischer ausdruck**

Exponentialfunktion mit Basis 10 und numerischer Ausdruck als Exponent.

TEN3 ist gleich 1000.

STRING-FUNKTIONEN

Ein String ist eine Kette aus alphanumerischen und Sonderzeichen. String-Funktionen sind eine Gruppe von Operationen, die Strings manipulieren. Sie nehmen einen String und errechnen einen numerischen Wert dazu, oder Sie nehmen einen numerischen Wert und errechnen den String dazu, oder Sie bearbeiten den String als String. In vielen BASIC-Dialekten muß das Argument der Funktion in Klammern gesetzt werden. Nicht so bei diesem Computer. Klammern müssen nur dann gesetzt werden, wenn es nötig ist, das Argument von anderen Werten deutlich zu trennen. String-Ausdrücke mit zwei oder drei Ausdrücken benötigen immer eine Klammerung.

ASC

1. ASC **"string ausdruck"**

Errechnet den Zeichen-Code des ersten Zeichens des Ausdrucks. Die Zeichen-Code-Tabelle finden Sie in Anhang B.

ASC "A" ist gleich 65.

Der Computer benutzt ASCII-Codes und ihre Zeichen.

CHR$

1. CHR$ **numerischer ausdruck**

Errechnet das Zeichen des gegebenen numerischen Wertes. CHR$ ist die Umkehrfunktion der ASC-Funktion.

CHR$ 65 ist gleich "A".

Achtung: Wenn der Zeichencode von 13 angegeben wird, während Sie manuell das CHR$-Kommando ausführen, wird der jeweils folgende Inhalt nicht angezeigt.
Beispiel:
CHR$70+CHR$71+CHR$13+CHR$75+CHR$76
ENTER → FG

Die Zeichen K und L für die Codes 75 und 76 werden nicht angezeigt.

LEFT$

1. LEFT$ ("**string ausdruck**", **numerischer ausdruck**)

Diese Funktion erstellt einen Teil-String des durch "**string ausdruck**" angegebenen Strings. Von links beginnend wird die mit **numerischer ausdruck** gegebene Anzahl von Zeichen herausgenommen.

LEFT$ ("ABCDEF", 2) ist gleich "AB".

LEN

1. LEN "**string ausdruck**"

Errechnet die Länge eines Strings.

LEN "ABCDE" ist gleich 5.

MID$

1. MID$ ("**string ausdruck**", **num asdr1, num asdr2)**

Diese Funktion erstellt einen Teil-String aus dem durch **ausdruck** angegebenen string. Durch **num asdr1** wird das erste Zeichen bestimmt, **num ausdr2** bestimmt die Anzahl der herauszunehmenden Zeichen.

MID$ ("ABCDEF", 2, 3) ist gleich "BCD".

RIGHT$

1. RIGHT$ ("**string ausdruck**", **numerischer ausdruck**)

Diese Funktion erstellt einen Teil-String des durch "**string ausdruck**" angegebenen Strings. Von rechts beginnend wird die Anzahl der von **numerischer ausdruck** gegebenenen Zeichen herausgenommen.

RIGHT$ ("ABCDEF", 3) ist gleich "DEF".

STR$

1. STR$ **numerischer ausdruck**

STR$ wandelt einen numerischen Wert in einen String um.

STR$ 1.59 ist gleich "1.59".

VAL

1. VAL "**string ausdruck**"

VAL ist die Umkehrfunktion zu STR$. VAL wandelt einen String in einen numerischen Ausdruck um.

VAL "1.59" ist gleich 1.59.

Achtung: VAL kann nur numerische Zeichen (Ø bis 9), Vorzeichen (+ und –) und das "E" für Exponenten umwandeln! Sind andere Zeichen in dem string ausdruck, so werden alle Zeichen rechts des letzten erlaubten Zeichens ignoriert. Spaces werden wie nicht-existent behandelt.

10.4 EIN-/AUSGABE-KOMMANDOS

CLOSE

1. CLOSE

Abkürzung: CLOS.
Vergleiche: OPEN

Wirkung:
Desaktiviert das serielle I/O-Interface.

Anwendung:
Das Kommando CLOSE schließt die Leitung, die für Ein-/Ausgabe zu seriellen Interface mit OPEN geöffnet wurde.

Nach diesem Kommando können Daten über das Interface weder empfangen noch gesendet werden.

CONSOLE

1. CONSOLE **ausdruck**

Abkürzung: CONS., CONSO., CONSOL.
Vergleiche: OPEN, LPRINT, LLIST

Wirkung:
Setzt bei Datensendung die Zahl der Zeichen pro Zeile.

Anwendung:
CONSOLE setzt für die Kommandos LLIST und LPRINT die Zahl der Zeichen pro Zeile fest, wenn Daten über das serielle Interface gesendet werden.

* Der Wert für **ausdruck** muß ganzzahlig sein und darf nicht kleiner als 1 oder größer als 160 sein. Wird **ausdruck** größer 160 gesetzt, so wird die Zahl der Zeichen pro Zeile auf 160 begrenzt.

  Wird **ausdruck** kleiner 1 gesetzt, erhalten Sie die Fehlermeldung ERROR 3.

  Wenn Sie im Kommando CONSOLE keinen Wert angeben, wird das Kommando ignoriert und die alte Zahl der Zeichen pro Zeile benutzt.

* Der voreingestellte Wert ist 39 Zeichen pro Zeile, der auch wieder eingenommen wird, wenn die Batterien des Rechners gewechselt werden.

INPUT#1

1. INPUT#1 **variable, variable, variable, variable,** . . .

Abkürzung: I.#1, IN.#1, INP.#1, INPU.#1
Vergleiche: OPEN, PRINT#1

Wirkung:
Lädt Daten über das serielle Interface in den Computer und weist sie den definierten Variablen zu.

Anwendung:

* Das Kommando wird nur ausgeführt, wenn die Leitung zum seriellen Ein-/Ausgabe-Interface vorher mit dem OPEN-Kommando geöffnet wurde. Ansonsten wird das Kommando ignoriert.

* Das INPUT# 1-Kommando erwartet Daten und weist sie den angegebenen Variablen zu. Dis Variablen müssen also dem Datentyp entsprechen. Vergleichen Sie hierzu mit dem PRINT#1-Kommando.

Beispiel:
INPUT# 1A, AB, C$, E(*)

Die empfangenen Daten werden den Variablen A, AB, C$ und dem Feld E( ) zugewiesen.

* Achten Sie darauf, daß die empfangenen Daten mit dem Variablentyp übereinstimmen, also Zahlen numerischen Variablen und Strings String-Variablen zugewiesen werden.

Wird im ASCII-Codesystem ein Zeichen einer numerischen Variablen zugewiesen, so wird ihr Wert gleich null gesetzt. Wird eine Zahl einer String-Variablen zugewiesen, so wird die Zahl in einen String umgewandelt. Sind die Daten und die Variablen nicht gleichen Typs, werden Sie also ungewollte Werte auf Ihren Variablen führen.

Selbst, wenn Daten in der Form einer Funktion, wie z.B. "SIN30", einer numerischen Variablen zugewiesen werden, wird die Variable gleich null gesetzt.

Bei Daten der Form "10+40" werden Zeichen nach dem Operationssymbol ignoriert, in diesem Fall wird nur "10" erfaßt.

Beachten Sie:

1. Enthalten die Engabedaten "CR" (Kontrollcode ØDH) oder "NULL" (Ø), können alle diesem Zeichen folgenden Daten ignoriert werden.
2. Einfache und Feld-Variable müssen im Programm-/Datenspeicherbereich Raum zugewiesen bekommen, bevor das INPUT#1-Kommando ausgeführt wird. Sonst bekommen Sie eine Fehlermeldung.

LLIST

1. LLIST
2. LLIST A
3. LLIST **ausdruck1, ausdruck2**

A ist "**label**" oder **ausdruck**

Abkürzung: LL., LLI., LLIS.
Vergleiche: OPEN, CONSOLE

Wirkung:
Sendet Programm-Listings über das serielle Interface.

Anwendung:
Dans LLIST-Kommando kann sowohl im PROGRAMM-Modus als auch im RUN-Modus eingesetzt werden.

Ist das Ein-/Ausgabe-Interface vorher mit dem OPEN-Kommando geöffnet worden, sendet das LLIST-Kommando das gespeicherte Programm als Listing im ASCII-Code an angeschlossene Geräte.

Ist die Leitung zum Interface geschlossen, wird das Listing über den Drucker ausgegeben.

* Form 1 sendet alle gespeicherten Programm-Listings.

Beispiel:
Geben Sie folgendes Demoprogramm ein:

```
  1Ø OPEN
 1ØØ REM **ABC-12**
65279 END
```

Geben Sie dann das Kommando:

LLIST [ENTER]

Das Listing wird auf dem Drucker ausgegeben. Geben Sie jetzt das Kommando:

OPEN [ENTER]
LLIST [ENTER]

Das Listing wird jetzt über das Interface ausgegeben.

* Form 2 sendet nur die Zeile, die mit dem Wert von **ausdruck** oder dem eingegebenen "**label**" gekennzeichnet ist.

* Form 3 sendet alle Zeilen, die zwischen den mit **ausdruck1** und **ausdruck2** benannten Zeilen liegen (inklusiv). **ausdruck1** und **ausdruck2** können auch durch "**label**" ersetzt werden.

* Wird in Form 3 **ausdruck1** weggelassen, so werden alle Zeilen von der ersten bis zu der mit **ausdruck2** bezeichneten Zeile ausgegeben.

* Wird in Form 3 **ausdruck2** weggelassen, so werden alle Zeilen ab der mit **ausdruck1** bezeichneten Zeile bis zu letzten ausgegeben.

* Gibt es keine Zeile mit der Nummer des **ausdruck1** oder **ausdruck2**, so wird jeweils die nachsthöhere Zeile genommen. Ist die Zeile von **ausdruck1** und **ausdruck2** die gleiche, erhalten Sie die Fehlermedlung ERROR 1.

* Wurde ein Passwort gesetzt, wird das Kommando LLIST ignoriert.

* Wurden weitere Programme mit MERGE eingeladen, so wird mit LLIST nur das letzte geladene Programm ausgegeben. Wollen Sie die vorher geladenen Programme ausgeben lassen, müssen Sie das Kommando in dieser Form benutzen:
  LLIST **"label"**

* Die Zahl der Zeichen pro Zeile wird mit dem CONSOLE-Kommando festgelegt. Wurde diese Zahl kleiner oder gleich 23 gesetzt, erhalten Sie die Fehlermeldung ERROR 3.

* Vor der Aussendung wird "$\sqrt{\ }$" in SQR umgewandelt.

LOAD

1. LOAD

Abkürzung: LOA.
Vergleiche: OPEN, CLOAD

Wirkung:
Lädt Daten über das serielle Ein-/Ausgabe-Interface in den Programm-/Datenspeicherbereich.

Anwendung:
Das Kommando wird nur ausgeführt, wenn die Leitung zum seriellen Interface vorher mit dem OPEN-Kommando geöffnet wurde. Ansonsten wird es ignoriert.

* Der Rechner lädt so lange Daten über das Interface ein, bis der Endcode erreicht ist. Dieser Datenblock wird als eine Programmzeile interpretiert. Der Computer wandelt die Daten dann in eine Form um, die er als Programm in seinem Programm-/Datenspeicherbereich ablegen kann. Dann liest er den nächsten Datensatz bis zum nächsten Endcode ein.
  Dieser Vorgang wiederholt sich, bis der Textendecode erreicht ist. Vergleichen Sie dazu mit dem OPEN-Kommando.

* In einem Datensatz können bis zu 256 Byte gelesen werden. Sollen also mehr als 256 Byte empfangen werden, erhalten Sie eine Fehlermeldung.

* Die eingelesenen Daten werden umgeformt und im Speicher abgelegt. Ist eine Programmzeile inklusiv Zeilennummer länger als 80 Byte, so erhalten Sie eine Fehlermeldung.

  Außerdem muß der Anfang einer Zeile immer ein numerischer Wert, also eine Zeilennummer, sein.

* Während der Ausführung des LOAD-Kommandos werden die Zeilen nicht neu sortiert (in aufsteigender Folge).

Beachten Sie:

* Die Ausführung des LOAD-Kommandos endet, wenn der Textendecode gelesen wird.

  Wird der Code nicht gesendet, müssen Sie die Ausführung wie folgt beenden:

1. Senden Sie nach der beendeten Übertragung des Programmes vom sendenden Gerät einen Textendecode.
2. Oder drücken Sie die BRK -Taste, um die Übertragung zu beenden.

Zwischencodes

* Daten werden als Zwischencodes gelesen und in den Programm/Daten-Bereich geschrieben.

* Wenn die Ausführung von LOAD aufgrund eines Fehlers unterbrochen wird, ist das gesamte Programm gelöscht. (Variable bleiben erhalten.)

LPRINT

1. LPRINT A
2. LPRINT A,A, . . . , A
3. LPRINT A;A; . . . ; A
4. LPRINT . . . A; Erweiterung der Formen 1, 2, 3 um ein Semikolon am Ende des Kommandos
5. LPRINT

A ist **ausdruck** oder **"string"**

Abkürzung: LP., LPR., LPRI., LPRIN.
Vergleiche: OPEN, CONSOLE, USING

Wirkung:
Sendet Daten über das serielle Ein-/Ausgabe-Interface.

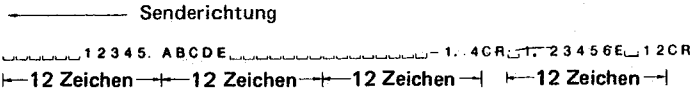
Anwendung:
Wurde die Leitung zum seriellen Interface mit dem OPEN-Kommando geöffnet, so werden die Daten im ASCII-Code übertragen. Ist die Leitung geschlossen, werden die Daten auf dem Drucker ausgegeben.

* Form 1 sendet den Wert des Ausdrucks oder die Zeichenkette (String). Ist der Wert des Ausdrucks negativ, wird ihm ein Minuszeichen (–) vorangesetzt. Ist er positiv, ein Leerfeld.
* Form 2 setzt automatisch 12 Zeichen pro Einheit fest und sendet die Daten in dieser Form.

Beispiel:

```
10 OPEN "1200, N, 8, 1, A, C"
20 CONSOLE 36
30 LPRINT 12345, "ABCDE", –7/5, 1.23456789E12
```

Bei Ausführung werden die Daten in folgender Form gesendet:


Senderichtung
␣␣␣␣␣12345. ABCDE␣␣␣␣␣

Das erste "CR" (Carriage Return – Zeilenende) wurde zwischengeschoben, da die Zeilenlänge mit dem Kommando CONSOLE 36 auf 36 Zeichen festgesetzt wurde. Das zweite "CR" kennzeichnet das Ende des Datensatzes.

Wenn ein String länger als 12 Zeichen ist, so werden nur die ersten 12 Zeichen des Strings gesendet.

Ebenso, wenn der Wert eines Ausdrucks mehr als 12 Stellen hat. Die Zahl wird in Exponentialdarstellung übertragen, die niederwertigen Stellen hinter dem Dezimalpunkt werden nicht gesendet. Vor der Zahl wird immer ein Leerfeld gesendet, wenn die Zahl positiv ist. Ist sie negativ, wird ein Minuszeichen vorangesetzt.

* Form 3 verändert nicht die Länge der zu sendenden Daten. Einer positiven Zahl wird kein Leerfeld vorangestellt.

Beispiel:

5Ø LPRINT –123; "ABC"; 567.89

Gesendet wird:

–123.ABC567.89CR

Der Datensatz wird, je nach OPEN-Modus, mit "CR" (Carriage Return – Wagenrücklauf) oder "LF" (Line Feed – Zeilenvorschub) beendet.

* Form 4 sendet keinen Datensatzendcode. Es wird nur der durch das CONSOLE-Kommando gesetzte "CR"-Code übertragen.

* Form 5 sendet nur den Datensatzendcode.

* Wurde das Format mit dem USING-Kommando vorgegeben, werden die Daten in allen Formen des Kommandos LPRINT entsprechend diesem USING übertragen.

* Die Ausführung von PRINT = LPRINT läßt den Rechner alle PRINT-Kommandos als LPRINT-Kommandos interpretieren. Das Kommando PRINT = LPRINT ist nur erlaubt, wenn der Drucker angeschlossen oder die Leitung zum seriellen Ein-/Ausgabe-Interface geöffnet ist.

Achtung:

Um Zeichen oder Kontrollcodes zu senden, die nicht auf der Tastatur vorhanden sind, müssen Sie das CHR$-Kommando benutzen.

Sie wollen das Technische AT (@) senden:

(1)
```
.
.
.
5Ø LPRINT CHR$ &40
```

(2)
```
.
.
.
5Ø A$=CHR$&40
6Ø LPRINT A$
```

Der Kontrollcode "NULL" (Ø) darf nur in Form 1 gesendet werden. In Form 2 wird er ignoriert.

OPEN

1. OPEN "baud rate, parität, wortlänge, stop bit, code typ, end code, textende code"
2. OPEN

Abkürzung: OP., OPE.
Vergleiche: CLOSE

Wirkung:

Öffnet die Leitung zum seriellen Ein-/Ausgabe-Interface und ermöglicht somit den Datentransfer über das Interface, setzt die dafür nötigen Parameter.

Anwendung:

Beide Formate öffnen die Leitung zum Interface.

* Form 1 erlaubt die Änderung der Parameterwerte in der angegebenen Reihenfolge. Jeder ausgelassene Parameterwert bis hin zur ersten Änderung muß durch ein Komma dargestellt werden. Ausgelassene Parameter behalten ihren aktuellen Wert.

* Form 2 benutzt die aktuellen Parameterwerte. Nach einem Batteriewechsel bzw. nach Betätigen des All-Reset-Schalters werden die Parameter auf die Standardwerte gesetzt.

  Die Standardwerte sind:

  "12ØØ, N, 8, 1, A, C, &1A"

Baud-Rate	12ØØ
Parität	N (none = keine)
Wortlänge	8 Bit
Stop-Bits	1
Codetyp	A (ASCII)
Endcode	C (Carriage Return)
Textende-Code	&1A (Dez 26)

* Jede der Optionen kann im OPEN-Kommando weggelassen werden. Der vorgegebene Wert wird dann eingesetzt.

Beispiel:

```
OPEN ",,,2"
```

Nur die Anzahl der Stop-Bits wird auf 2 gesetzt, alle anderen Werte bleiben unverändert.

Baud-Rate: 300, 600, 1200
Setzt die Übertragungsrate. Der Computer kann mit 300, 600 oder 1200 Baud Daten senden und empfangen (1 Baud = 1 Bit/sec).

Parität: N, E, O

Setzt den Typ der Parität. N: NONE – es wird kein Paritäts-Bit benutzt. E: EVEN – setzt gerade Parität. O: ODD – setzt ungerade Parität.

Wortlänge: 7, 8

Setzt die Zahl der Bit pro Zeichen. Es können 7 oder 8 Bit pro Zeichen gesetzt werden.

Stop-Bits: 1, 2

Setzt die Zahl der Stop-Bits.

Codetyp: A, B

Spezifiziert das zu übertragende und empfangende Code-System.
A: Spezifiziert ASCII-Codes.
B: Spezifiziert Zwischencodes.
Die Wortlänge kann für Zwischencodes nicht auf 7 Bits gesetzt werden, weil dann ERROR 1 entsteht.

Endcode: C, F, L

Setzt den Code für das Zeichen "Ende des Datensatzes". C: Setzt den Code CR (Carriage Return = Wagenrücklauf). F: Setzt den Code LF (Line Feed = Zeilenvorschub). L: Setzt den Code CR + LF.

Textendecode: &00 bis &FF (Dez 0 bis 255)

Setzt den Code für das Zeichen "Ende des Textes", um z. B. das Ende eines Programmes erkennen zu können. Kann bei den Kommandos SAVE und LOAD gefordert sein.

* Wird das OPEN-Kommando gegeben, obwohl das serielle Interface bereits aktiviert war, wird Error 8 angezeigt.

Mit dem Kommando CLOSE wird das Interface desaktiviert, ebenso wie mit dem RUN-Kommando, einem Programm-Ende, dem Einschalten des Computers oder der Umschaltung in den CAL-Modus. Die aktuellen Parameterwerte bleiben dabei erhalten.

OPEN$

1. OPEN$

Abkürzung: OP.$, OPE.$
Vergleiche: OPEN

Wirkung:
Die aktuellen Parameterwerte des seriellen Interface werden angezeigt.

Beispiel:

OPEN$ [ENTER]

Ausgabe:

1200, N, 8, 1, A, C, &1A

Hinweis: Dieses Kommando dient zum Überprüfen der E/A-Bedingungen und kann daher nicht in Kombination mit einer anderen Funktion verwendet werden. Bei Verwendung dieses Kommandos in einem Programm wird es als PRINT OPEN$ geschrieben.

PRINT#1

1. PRINT#1 **variable, variable, variable,** . . .

Abkürzung: P.#1, PR.#1, PRI.#1, PRIN.#1
Vergleiche: OPEN, INPUT#1, PRINT#

Wirkung:
Sendet Daten über das serielle Ein-/Ausgabe-Interface.

Anwendung:
Das Kommando wird nur ausgeführt, wenn das Interface mit OPEN aktiviert wurde. Sonst wird es ignoriert.

* Folgende Variablentypen können gesendet werden:

Im ASCII-Code
Standardvariablen (Konstante)
Jeder Variablenname muß angegeben werden.

Beispiel: A, B, C$

Achtung: Diese Variablen können nicht die Form A* haben.

Einfache Variable
Jeder Variablenname muß angegeben werden.

Beispiel: AA, B1$, C2

Feld-Variable (Arrays)
Müssen in der Form eines Feld-Namen (*) angegeben werden.

Beispiel: B(*), C$(*)

Alle Elemente eines Feldes werden in dieser Form übertragen. Einzelne Feld-Elemente können nicht direkt aus dem Feld gesendet werden.

Beispiel: 5Ø PRINT#1 A, AB, C$, E(*)

* Der Endcode wird an das Ende einer jeden gesendeten Variablen oder eines jeden gesendeten Feld-Elements angehängt.

Beispiel:

Wenn A=12345 und B$="ABC" ist, dann werden die Daten des Kommandos

PRINT#1A, B$

in dieser Form gesendet (Endcode sei CR):

1.2345ØØØØØEØ4"CR"ABC"CR"

* Ist der Wert einer numerischen Variablen negativ, so wird vor der Zahl ein Minuszeichen (–) gesendet.

* Die Elemente eines Feldes werden in dieser Form übertragen:

Beispiel 1:

Eindimensionales Feld B(3): B(Ø) → B(1) → B(2) → B(3)

Beispiel 2:

Zweidimensionales Feld (2, 3): C(Ø,Ø) → C(Ø, 1) → C(Ø,2) → C(Ø,3) → C(1,Ø). . . .

Im Zwischencode

* Die Spezifikation dieser Variablen ist identisch mit den im ASCII-Code gesendeten Variablen mit der Ausnahme, daß Standardvariablen auch in der Form A* spezifiziert werden können. (Siehe PRINT #.)

* Variable werden entsprechend dem internen Format gesendet, aber der Endcode wird nicht gesendet. Dem Anfang jeder Variablen wird ein Label (Gesamtzahl der gesendeten Bytes, STRING Element etc.) vorangestellt. Wenn Variable im A* Format spezifiziert sind, wird am Ende der CR-Code gesendet.

Achtung:

* Erweiterte Variable, wie A(27) oder darüber sowie einfache und Feld-Variable, müssen initialisiert sein, bevor sie gesendet werden können. Im anderen Fall erhalten Sie ein Fehlermeldung.

  Sie erhalten auch eine Fehlermeldung, wenn der Typ der Variablen im Computer nicht mit dem Typ der zu sendenden übereinstimmt.

* Sind in Daten Sonderzeichen, wie z.B. das Wurzelzeiche, enthalten, werden sie in die entsprechende Kurzform, hier SQR, umgewandelt und dann gesendet.

SAVE

1. SAVE

Abkürzung: SA., SAV.
Vergleiche: OPEN, LLIST

Wirkung:
Sendet Programme aus dem Computer über das serielle Ein-/Ausgabe-Interface.

Anwendung:
Das Interface muß mit dem OPEN-Kommando aktiviert werden, sonst wird das SAVE-Kommando ignoriert.

* Das SAVE-Kommando wird nicht ausgeführt, wenn ein Passwort gesetzt wurde.

* Zwischen den beiden Code-Systemen bestehen die folgenden Unterschiede.

ASCII-Code: Sendet am Ende jeder Zeile einen Endcode. Sendet am Ende einer Textübertragung einen Text-Endcode.

Zwischencode: Sendet Endcodes und Text-Endcodes.

10.5 TEXT-FUNKTIONSKOMMANDOS

BASIC

1. BASIC

Abkürzung: BA., BAS., BASI.
Vergleiche: TEXT

Wirkung:
Löscht den TEXT-Modus, nur in der Direkteingabe im PROGRAMM-Modus erlaubt.

Anwendung:
Die Ausführung des BASIC-Kommandos löscht den TEXT-Modus und schaltet den Computer in den BASIC-Modus zurück.

Das Bereitschaffssymbol wechselt von "<" zu ">".

* Der Wechsel aus dem TEXT- in den BASIC-Modus wandelt einen Text im Computer in ein Programm um (interner Code).

Alle Kleinbuchstaben mit Ausnahme der in Zeichenstrings in Anführungsstrichen (" ") eingeschlossenen und der in REM-Anweisungen stehenden werden automatisch in Großbuchstaben umgewandelt.

Abkürzungen, wie z.B. "P." oder "I.", werden nicht in die entsprechenden Kommandos, hier PRINT bzw. INPUT, übersetzt. Führen Sie in diesem Fall den Cursor in die entsprechende Zeile und drücken Sie dann [ENTER] . Kommandos und Formate, die nicht im Computer vorgesehen sind, können nicht ausgeführt werden.

* Während der Umformung erskeint am rechten Ende des Bildschirms "**".

* Wurde ein Passwort gesetzt, erhalten Sie auf das Kommando BASIC die Fehlermeldung ERROR 1.

TEXT

1. TEXT

Abkürzung: TE., TEX.
Vergleiche: BASIC

Wirkung:
Schaltet den Computer in den TEXT-Modus, nur erlaubt in Direkteingabe im PROGRAMM-Modus.

Anwendung:
Die TEXT-Funktion (oder -Modus) wird gebraucht, wenn ein Programm in den Computer eingegeben werden soll, das für einen Personalcomputer geschrieben wurde. Die Eingabe kann auch über das serielle Ein-/Ausgabe-Interface erfolgen.

* Die Ausführung des TEXT-Kommandos schaltet den Computer in den TEXT-Modus. Daten, bestehend aus einer Zahl, der Zeilennummer und die Programminformation, Befehl und Kommandos werden in den Rechner eingegeben. Dann wird die [ENTER] -Taste gedrückt, um die Daten in den Programm-/Datenspeicherbereich zu übertragen.

Natürlich können Befehle, die nicht im Computer vorgesehen sind, auch nicht in den internen Code des Rechners übertragen werden.

Der Text wird so, wie er eingegeben wurde, als Folge von ASCII-Zeichen gespeichert. Er wird in der Reihenfolge der als Zeilennummern vorgesehenen Zahlen sortiert (Zeilennummer-Editierfunktion).

* Ein Text wird in der eingegebenen Form gespeichert.

Deshalb werden BASIC-Abkürzungen, wie z.B. "I." für "INPUT", so wie sie eingegeben werden, angezeigt und gespeichert.

* Ein Programm, das im internen Code des Computer gespeichert ist, wird bei Umschaltung in den TEXT-Modus in den ASCII-Code umgewandelt.

* Während der Programmumwandlung erscheint am rechten Rand "**".

* Das Bereitschaftssymbol des TEXT-Modus ist "<", das des BASIC-Modus ist ">".

* Inklusiv der Zeilennummer und dem [ENTER] darf eine Zeile im TEXT-Modus nicht länger als 80 Zeichen sein. Überzählige Zeichen werden gelöscht.

* Die Zahl der Bytes erhöht sich, wenn ein Programm aus dem internen Code in den ASCII-Code übertragen wird, da im internen Code ein Kommando, z.B. "PRINT", nur ein Byte, im ASCII-Zeichencode aber 5 Bytes lang ist. Wird durch die Umwandlung der Speicherbereich überschritten, wird die Umwandlung rückgängig gemacht und die Fehlermeldung ERROR 6 ausgegeben.

* Wurde ein Passwort gesetzt, wird auf das Kommando TEXT die Fehlermeldung ERROR 1 ausgegeben.

KAPITEL 11
FEHLERSUCHE

In diesem Kapitel sollen Sie einige Hinweise erhalten, was Sie unternehmen können, wenn Ihr Computer nicht tut, was Sie von ihm erwarten, Es ist in zwei Teile gegliedert — der erste beschäftigt sich mit allgemeiner Bedienung des Gerätes, und der zweite mit der BASIC-Programmierung. Für jedes Problem schlagen wir eine Reihe von Lösungen vor. Sie sollten jede von ihnen versuchen — aber nur eine zur Zeit —, bis Sie das Problem gelöst haben.

BEDIENUNG DES GERÄTS

Wenn:	**Dann sollten Sie:**
Sie das Gerat einschalten, aber nichts in der Anzeige erscheint.	1) Überprüfen, ob Sie das Gerät eingeschaltet haben (ON-Position). 2) Die [BRK] -Taste betätigen, um festzustellen, ob sich das Gerät automatisch abgeschaltet hatte. 3) Die Batterien wechseln. 4) Den Kontrast einstellen. 5) Überprüfen, ob sich der Schiebeschalter in der LOCK-Position befindet.
Die Anzeige zwar funktioniert, aber keine Reaktion auf Tastendruck erfolgt.	1) [C·CE] drücken, um die Anzeige zu löschen. 2) [CA] ([SHIFT] [C·CE]) drücken, um die Anzeige zu löschen. 3) Das Gerät aus- und wieder einschalten. 4) Irgendeine Taste drücken, festhalten und den ALL-RESET-Schalter drücken. 5) Den ALL-RESET-Schalter ohne zusätzlichen Tastendruck betätigen.
Sie eine Rechnung oder Antwort eingeben, aber keine Reaktion erfolgt.	1) Drücken Sie [ENTER] .
Sie ein BASIC-Programm abarbeiten lassen, etwas angezeigt wird und das Programm dann anhält.	1) Drücken Sie [ENTER] .

Sie eine Rechnung eingeben und diese in der Form eines BASIC-Statements angezeigt wird (Doppelpunkt nach der ersten Zahl).	1) Schalten Sie vom PROgramm- in den RUN-Modus um.
Sie keinerlei Reaktion auf Tastenbetätigung erhalten.	1) Halten Sie irgendeine Taste fest und drücken Sie den ALL-RESET-Schalter. 2) Wenn Sie dann noch immer keine Reaktion erhalten, drücken Sie den ALL-RESET-Schalter, ohne dabei eine andere Taste festzuhalten. Dabei werden jedoch alle Daten, Programme und Speicherinhalte gelöscht.

FEHLERSUCHE IM BASIC

Wenn Sie ein neues BASIC-Programm eingeben, wird dieses in der Regel beim ersten Startversuch nicht laufen. Selbst, wenn Sie nur ein Programm abtippen, von dem Sie wissen, daß es korrekt ist, wie z.B. die in diesem Handbuch vorgestellten, dürfte Ihnen normalerweise mindestens ein Tippfehler unterlaufen. Handelt es sich um ein längeres Programm, wird es oft auch mindestens einen logischen Fehler enthalten. Es folgen einige grundsätzliche Hinweise, wie Sie solche Fehler finden und korrigieren.

Sie starten Ihr Programm und erhalten eine Fehlermeldung:

1. Schalten Sie zurück in den PROgramm-Modus und benutzen Sie die [↑] oder [↓] -Taste, um die fehlerhafte Zeile ins Display zu rufen. Der Cursor wird sich an der Stelle befinden, wo der Computer verwirrt wurde.

2. Wenn Sie aus der Art, wie die Programmzeile geschrieben ist, keinen offensichtlichen Fehler entnehmen können, kann das Problem auch an den von Ihnen verwendeten Werten liegen. So wird beispielsweise CHR$(A) eine Fehlermeldung bewirken, wenn A den Wert 1 hat, weil CHR$(1) ein unerlaubtes Zeichen ist. Überprüfen Sie die Werte der von Ihnen verwendeten Variablen, indem Sie entweder im RUN- oder im PROgramm-Modus die einzelnen Variablennamen gefolgt von [ENTER] eingeben.

Sie starten das Programm mit RUN und erhalten keine Fehlermeldung, doch das Programm tut nicht, was Sie von ihm erwarten:

3. Überprüfen Sie das Programm Zeile für Zeile unter Verwendung von LIST und den [↓] - und [↑]-Tasten, um herauszufinden, ob Sie das Programm korrekt

eingegeben haben. Es ist erstaunlich, wieviele Fehler beim bloßen erneuten Durchsehen eines Programms gefunden werden können!

4. Versuchen Sie, jede einzelne Zeile beim Durchsehen so zu interpretieren, als wären Sie ein Computer. Nehmen Sie einfache Werte und realisieren Sie die Operationen der einzelnen Zeilen, um herauszufinden, ob Sie die gewünschten Ergebnisse erhalten.

5. Fügen Sie eines oder mehrere zusätzliche PRINT-Statements in das Programm ein, um die einzelnen Werte und Tastenbelegungen zur Anzeige zu bringen. Benutzen Sie diese, um die korrekten Teile des Programms von den möglicherweise fehlerhaften zu isolieren. Diese Vorgehensweise ist auch nützlich um zu bestimmen, welche Teile des Programms schon abgearbeitet wurden. Sie können den Programmablauf auch an kritischen Stellen vorübergehend mit STOP unterbrechen, um dann einzelne Variable zu überprüfen.

6. Verwenden Sie TRON und TROFF, entweder als Befehle oder als Programmbestandteile, um den Programmablauf durch die einzelnen Zeilen hindurch verfolgen zu können. Halten Sie das Programm an kritischen Punkten an, um den Inhalt von wichtigen Variablen zu überprüfen. Dies ist zwar eine sehr langsame Methode, Fehler aufzuspüren – aber es ist manchmal die einzige.

KAPITEL 12
RICHTIGE BEHANDLUNG DES COMPUTERS

Um reibungsloses Funktionieren Ihres Computers zu gewährleisten, empfehlen wir die Beachtung der folgenden Punkte:

- Gehen Sie immer vorsichtig mit Ihrem Taschencomputer um, da die Flüssigkristallanzeige aus Glas gefertigt ist.
- Halten Sie den Computer von starken Temperaturschwankungen fern, ebenso von Feuchtigkeit oder Staub. Wenn Sie bei warmen Wetter Ihren PKW längere Zeit der direkten Sonne aussetzen und sich hohe Temperaturen aufbauen, kann das Ihren Computer beschädigen!
- Benutzen Sie zur Reinigung Ihres Computers ausschließlich ein trockenes, weiches Tuch. Verwenden Sie niemals Lösungsmittel, Wasser oder ein feuchtes Tuch!
- Um das Auslaufen der Batterien zu vermeiden, entfernen Sie diese, wenn Sie den Computer für längere Zeit nicht benutzen wollen.
- Wenn Sie die Hilfe einer Werkstatt benötigen, geben Sie Ihren Computer nur in ein autorisiertes SHARP Service Center oder wenden Sie sich an einen SHARP Fachhändler.
- Wenn der Computer starker statischer Aufladung oder auch starken Störungen ausgesetzt ist, kann er sich unter Umständen "aufhängen" (d. h. auf Tastendruck nicht mehr reagieren). Falls dies auftritt, drücken Sie den ALL-RESET-Schalter und halten dabei eine Taste fest (siehe Pannenhilfe).
- Bewahren Sie dieses Handbuch gut auf, falls Sie später einmal etwas nachschlagen wollen.

ANHANG

ANHANG A

FEHLERMELDUNGEN

Fehlernummer	Bedeutung
1	Syntax-Fehler

Der Computer kann nicht verstehen, was Sie eingegeben haben. Prüfen Sie die Eingabe auf Dinge wie Semikolon am Ende eines PRINT-Statements, falsch geschriebene Befehle und fehlerhaften Gebrauch von Zeichen.

(z.B.: 3 * /2)

2 Rechenfehler

Hier haben Sie vermutlich eines der drei folgenden Dinge getan:

1. Versucht, eine zu große Zahl zu benutzen; das Rechenergebnis ist größer als 9.999999999 E 99.
2. Versucht, durch Null zu teilen.
   (z.B.: 5 / Ø)
3. Versucht, eine unlogische Rechnung auszuführen.
   (z.B.: LN – 3Ø oder ASN 1.5)

3 Unzulässige Funktion (DIMensionierungs-Fehler, Argument-Fehler)

Die Feld-Variable existiert bereits. Sie haben versucht, ein Feld zu spezifizieren, ohne es zuvor zu dimensionieren.

Der Index des Feldes übersteigt die im DIM-Statement vorgegebene Größe.
(z.B.: DIM B(256))

Unzulässiges Funktionsargument, d.h. Sie haben versucht, Ihren Computer etwas tun zu lassen, wozu er nicht imstande ist.
(z.B.: WAIT 66ØØØ)

- Wenn der LLIST-Befehl für Ausgabe über die serielle E/A-Schnittstelle verwendet wird, sollte der CONSOLE-Wert mit dem CONSOLE-Befehl auf 23 Zeichen pro Zeile eingestellt sein.

- Wenn der CONSOLE-Wert auf 6 oder weniger Zeichen pro Zeile eingestellt ist, kann der PC-1450 für manuelles Drucken auf dem CE-140P verwendet werden.

4 Zu hohe Zeilennummer

Hier haben Sie vermutlich einen der folgenden Fehler gemacht:

1. Versucht, mit GOTO, GOSUB, RUN, LIST, THEN etc. eine nicht vorhandene Zeile anzusprechen.
2. Versucht, eine zu große Zeilennummer zu verwenden. Die höchstmögliche Zeilennummer ist 65279.

5 Verschachtelungsfehler

Das Einrichten einer Subroutine übersteigt 1Ø Ebenen.

Das Einrichten einer FOR-Schleife übersteigt 5 Ebenen.

Die Befehle RETURN ohne GOSUB, NEXT ohne FOR, READ ohne DATA wurden benutzt.

Fassungsvermögen des Puffers überschritten.

6 Speicherkapazität überschritten

Diese Fehlermeldung erhalten Sie normalerweise, wenn Sie versuchen, ein Feld zu DIMensionieren, das zu groß für die Speicherkapazität ist, oder wenn ein Programm zu lange ist.

7 PRINT USING Fehler

Das bedeutet, daß in einem USING-Statement eine unzulässige Format-Spezifikation enthalten ist.

8 E/A-Anschluß-Fehler

Dieser Fehler kann nur auftreten, wenn Sie den zusätzlichen Drucker und/oder Cassettenrecorder an den Computer angeschlossen haben.

Dieser Fehler kann auch auftreten, wenn Sie den seriellen E/A-Anschluß benutzen.

Die Meldung informiert Sie über ein Übermittlungsproblem zwischen dem Drucker und/oder Cassettenrecorder und dem Computer.

9 Andere Fehler

Dieser Fehlercode wird angezeigt, wenn der Computer ein Problem hat, das mit den anderen Fehlercodes nicht zu erfassen ist. Eine der häufigsten Ursachen für das Auftreten dieser Meldung ist der Versuch, Daten einer Variablen unter einem bestimmten Namen anzusprechen (z.B. A$), während die Daten der Variablen unter dem anderen Namen (d.h. A) abgespeichert worden waren.

RENUM BETREFFENDE FEHLER

Fehlermeldung	Beschreibung
ERROR 1	Der RENUM-Befehl enthält einen Syntaxfehler.
ERROR 1 IN *Zeilennummer*	Im Befehl für das Sprungziel fehlt eine Zeilennummer (z.B. GOTO, GOSUB usw.).
ERROR 3 IN Zeilennummer	Während der Ausführung eines RENUM-Befehls wird eine Zeilennummer größer als 65279 angetroffen. Die Länge einer Programmzeile überschreitet 79 Bytes.
ERROR 4	Die angegebene alte Zeilennummer ist nicht im Programm vorhanden.
ERROR 4 IN Zeilennummer	Die als Sprungziel angegebene Zeilennummer in der Programmdatei nicht vorhanden.
ERROR 6	Die Speicherkapazität reicht nicht zur Durchführung des RENUM-Befehls oder wird während der Neunumerierung rung reduziert.
ERROR 9	Es wurde versucht, RENUM in einer anderen Betriebsart als PRO (Programm) durchzuführen. Es wurde versucht, die Ablauffolge von Programmzeilen zu ändern, indem für die neue Zeilennummer ein niedrigerer Wert vorgegeben wurde als die Zeilennummer unmittelbar vor der alten Zeilennummer.
ERROR 9 IN Zeilennummer	Die als Sprungziel angegebene Zeilennummer ist ungeeignet, weil sie eine Variable, einen Ausdruck oder eine Funktion verwendet (z.B. unkorrekte Zeilennummern-Bezugnahme).

ANHANG B ASCII-CODE TABELLE

Die folgende Tabelle enthält die Umwandlungswerte für CHR$ und ASC. Die Spalten geben das erste hexadezimale Zeichen bzw. die ersten vier Bits, die Zeilen das zweite hexadezimale Zeichen bzw. die letzten vier Bits. Oben links in jedem Kästchen ist der Dezimalcode für das jeweilige Zeichen angegeben. Das Zeichen selbst befindet sich unten rechts. Wenn kein Zeichen angegeben ist, ist dieses Zeichen nicht verfügbar. Beispielsweise ist das Zeichen "A" dezimal 65, hexadezimal 41 und binär Ø1ØØØØØ1. Das Zeichen "√" ist dezimal 252, hexadezimal FC und binär 11111ØØ.

Hinweis:

* Die Zeichen für die Zeichencodes 92 (&5C), 249 (&F9) und 250 (&FA) unterscheiden sich in der Anzeige auf dem Display vom Ausdruck mit den Druckern CE-126P und CE-140P.

* Das Zeichen für den Zeichencode 92 (&5C) wird auf dem Drucker CE-126P als ◣ und auf dem Drucker CE-140P als \ ausgegeben.

* Bei Verwendung des Druckers CE-126P darf der Code 0 (&00) keinesfalls verwendet werden.

Erste 4 Bits

hex binär	Ø ØØØØ	1 ØØØ1	2 ØØ1Ø	3 ØØ11	4 Ø1ØØ	5 Ø1Ø1	6 Ø11Ø	7 Ø111	8 1ØØØ	E 111Ø	F 1111
Ø ØØØØ	Ø NUL	16	32 SPACE	48 Ø	64 @	8Ø P	96 `	112 p	128	224	24Ø
1 ØØØ1	1	17	33 !	49 1	65 A	81 Q	97 a	113 q	129	225	241
2 ØØ1Ø	2	18	34 "	5Ø 2	66 B	82 R	98 b	114 r	13Ø	226	242
3 ØØ11	3	19	35 #	51 3	67 C	83 S	99 c	115 s	131	227	243
4 Ø1ØØ	4	2Ø	36 $	52 4	68 D	84 T	1ØØ d	116 t	132	228	244
5 Ø1Ø1	5	21	37 %	53 5	69 E	85 U	1Ø1 e	117 u	133	229	245 ♠
6 Ø11Ø	6	22	38 &	54 6	7Ø F	86 V	1Ø2 f	118 v	134	23Ø	246 ♥
7 Ø111	7	23	39 '	55 7	71 G	87 W	1Ø3 g	119 w	135	231	247 ♦
8 1ØØØ	8	24	4Ø (	56 8	72 H	88 X	1Ø4 h	12Ø x	136	232	248 ♣
9 1ØØ1	9	25	41)	57 9	73 I	89 Y	1Ø5 i	121 y	137	233	249 ▉
A 1Ø1Ø	1Ø	26	42 *	58 :	74 J	9Ø Z	1Ø6 j	122 z	138	234	25Ø ▒
B 1Ø11	11	27	43 +	59 ;	75 K	91 [	1Ø7 k	123 {	139	235	251 π
C 11ØØ	12	28	44 ,	6Ø <	76 L	92 \	1Ø8 l	124 ¦	14Ø	236	252 √
D 11Ø1	13	29	45 –	61 =	77 M	93]	1Ø9 m	125 }	141	237	253
E 111Ø	14	3Ø	46 .	62 >	78 N	94 ^	11Ø n	126 ~	142	238	254
F 1111	15	31	47 /	63 ?	79 O	95 _	111 o	127	143	239	255

Zweite 4 Bits

ANHANG C

FORMATIEREN DER DATENAUSGABE

Es ist manchmal wichtig oder nützlich, neben dem Inhalt ausgegebener Daten auch das Format zu kontrollieren. Der Computer bedient sich hierzu des Befehls USING. Mit Hilfe dieses Befehls können Sie spezifizieren:

- die Anzahl der Stellen
- die Position des Dezimalpunktes
- wissenschaftliche Notation
- Anzahl der Zeichen in einem String

Diese verschiedene Formate bestimmen Sie mit einer "Ausgabe Maske", die aus einer String-Konstanten oder auch einer String-Variablen bestehen kann.

```
1Ø: USING "####"
2Ø: M$ = "&&&&&"
3Ø: USING M$
```

Wird der Befehl USING ohne Maske benutzt, werden damit alle Spezialformate aufgehoben.

```
4Ø: USING
```

Der Befehl USING kann auch in einem PRINT-Statement benutzt werden.

```
5Ø: PRINT USING M$; N
```

Wann immer USING benutzt wird, bewirkt es die Kontrolle aller ausgegebenen Daten, bis ein neuer USING-Befehl auftritt.

NUMERISCHE DATEN

Eine numerische USING-Maske darf nur benutzt werden, um numerische Werte, d.h. numerische Konstante oder numerische Variable zu kontrollieren. Wird eine String-Konstante oder -Variable angezeigt, während eine numerische USING-Maske wirksam ist, wird die Maske hierauf nicht angewendet werden.

Ein auszugebender Wert muß immer in den von der Maske vorgesehenen Freiraum passen. Es muß auch Raum für das Vorzeichen vorgesehen sein, selbst, wenn die Zahl immer positiv sein wird. Eine Maske, die vier Stellen zuläßt, kann also nur für dreistellige Zahlen verwendet werden.

NUMERISCHE MASKEN

Die gewünschte Anzahl von Stellen wird mit Hilfe des "#"-Zeichens festgelegt. Die Anzeige oder der Ausdruck beinhalten immer soviele Zeichen, wie in der Maske vorgesehen sind. Dabei erscheint die Zahl rechts in diesem Bereich, die restlichen Stellen werden durch Leerschritte aufgefüllt. Bei positiven Zahlen wird sich daher auf der linken Seite der Anzeige immer mindestens ein Leerschritt befinden. Da der Computer nicht mehr als 10 Stellen erfassen kann, sollte die größte verwendete numerische Maske maximal 11 "#"-Zeichen enthalten.

Wenn die Gesamtstellenzahl des ganzzahligen Teils 11 Stellen übersteigt, wird dieser ganzzahlige Teil als 11 Stellen aufgefaßt.

Anmerkung: In allen Beispielen dieses Anhangs werden Sie am Anfang und am Ende eines Anzeigefeldes "|"-Zeichen finden, mit denen die Größe des Feldes anschaulich gemacht werden soll.

EINGABE	ANZEIGE
10: USING "####"	(Bringen Sie den Computer in den RUN-Mode, geben Sie RUN ein und drücken Sie die [ENTER] -Taste.)
20: PRINT 25	| 25|
30: PRINT -350	| -350|
40: PRINT 1000	ERROR 7 in 40

Beachten Sie, daß die letzte Eingabe eine Fehlermeldung hervorruft, weil 5 Stellen (4 Ziffern und eine Stelle für das Vorzeichen) benötigt werden, die Maske aber nur 4 vorsieht.

NACHKOMMASTELLEN

Das Zeichen für den Dezimalpunkt (".") kann in einer Maske enthalten sein, um die gewünschte Position des Dezimalpunktes festzulegen. Wenn die Maske mehr Stellen bereitstellt, als für den anzuzeigenden Wert benötigt werden, werden die auf der rechten Seite übrigbleibenden Stellen mit Nullen aufgefüllt. Enthält der anzuzeigende Wert mehr Stellen, als die Maske vorsieht, werden diese abgeschnitten (nicht gerundet).

EINGABE	ANZEIGE
10: USING "#####.##"	
20: PRINT 25	| 25.00|
30: PRINT -350.5	|-350.50|
40: PRINT 2.547	| 2.54|

WISSENSCHAFTLICHE NOTATION

Das "^"-Zeichen kann in einer Maske enthalten sein, um so anzuzeigen, daß die Zahl in wissenschaftlicher Notation ausgegeben werden soll. Die Zeichen "#" und "." werden in der Maske benutzt, um den "charakteristischen" Teil der Zahl (d.h. den Teil, der links vom "E" steht) zu formatieren. Links vom Dezimalpunkt sollten immer 2 "#"-Zeichen stehen, um genung Raum für eine ganze Zahl und das Vorzeichen zu schaffen. Der Dezimalpunkt kann – muß aber nicht – vorgegeben werden.

Rechts vom Dezimalpunkt können bis zu 9 Stellen eingerichtet werden (mit Hilfe von "#"-Zeichen). Nach dem "charakteristischen" Teil wird das Exponentiationszeichen (E) angezeigt, gefolgt von der Stelle für das Vorzeichen und dem Exponenten (2stellig). Das kleinste von einer Maske vorzugebende Feld für wissenschaftliche Notation wäre demnach "##^", damit werden Zahlen in der Form "2 E 99" dargestellt. Das größtmögliche Feld für wissenschaftliche Notation wäre "##.#########^", das Zahlen wie "–1.234567890 E –12" darstellen kann.

EINGABE	ANZEIGE
10: USING "###.##^"	
20: PRINT 2	| 2.00E 00 |
30: PRINT –365.278	|–3.65E 02 |

ALPHANUMERISCHE MASKEN

String-Konstanten und -Variable werden mit Hilfe des "&"-Zeichens ausgegeben. Jedes "&" gibt eine im vorgesehenen Feld anzuzeigende Stelle an. Der String wird auf der linken Seite dieses Feldes angezeigt. Ist der String kürzer als der dafür eingeräumte Raum, werden die rechts verbleibenden Stellen mit Leerschritten aufgefüllt. Ist der String länger als das Feld, wird er abgeschnitten:

EINGABE	ANZEIGE
10: USING "&&&&&&"	
20: PRINT "ABC"	| ABC |
30: PRINT "ABCDEFGHI"	| ABCDEF |

GEMISCHTE MASKEN

In den meisten Anwendungsfallen wird eine USING-Maske entweder alle notwendigen Zeichen zur Formatierung eines numerischen Ausdrucks oder zur Formatierung eines Strings beinhalten. Für bestimmte Zwecke können aber auch beide gemeinsam in einer USING-Maske enthalten sein. In solchen Fällen markiert jede Umschaltung von String-formatierenden Zeichen auf Zahlen-formatierende

Zeichen (und umgekehrt) die Grenze für einen bestimmten Wert. So können mit einer Maske der Form "#####&&&&" zwei verschiedene Werte formatiert werden – ein numerischer Ausdruck, für den 5 Stellen vorgesehen sind, und ein alphanumerischer Ausdruck, für den 4 Stellen bereitgestellt wurden.

EINGABE	ANZEIGE
1Ø: PRINT USING "###.##&&"; 25; "CR"	| 25.ØØCR |
2Ø: PRINT –5.789; "DB"	| –5.78DB |

Achtung: Wurde ein USING-Statement einmal spezifiziert, wirkt es sich auf alle nachfolgenden Daten aus, bis es aufgehoben oder durch einen anderen USING-Befehl ersetzt wird.

ANHANG D

BEWERTUNG VON AUSDRÜCKEN UND OPERATOREN-VORRANG

Wenn in den Computer ein komplexer Ausdruck eingegeben wird, bewertet er Teile dieses Ausdrucks in einer Reihenfolge, die durch die Vorrangstellung der einzelnen Teile bestimmt wird. Geben Sie den Ausdruck

```
100/5+45
```

ein, entweder als Rechenoperation oder als Teil eines Programms, so weiß der Computer nicht, ob Sie meinen:

$$\frac{100}{5+45} = 2 \quad \text{oder} \quad \frac{100}{5} + 45 = 65$$

Da der Computer eine Möglichkeit haben muß, zwischen diesen beiden Operationen zu entscheiden, bedient er sich seiner Regeln des Operatoren-Vorrangs. Da die Division eine höhere "Priorität" hat als die Addition (siehe unten), wird er entscheiden, daß zuerst die Division durchgeführt wird und anschließend die Addition, d.h. die zweite Möglichkeit wird ausgeführt und als Ergebnis 65 ausgegeben.

VORRANG VON OPERATOREN

Im BASIC-Modus werden Operatoren gemäß den folgenden Prioritätsregeln verarbeitet (angefangen mit der höchsten Priorität):

1. Klammern
2. Variable und Pseudovariable
3. Funktionen
4. Exponentiation (^)
5. Vorzeichen (+, –)
6. Multiplikation und Division (*, /)
7. Addition und Sutraktion (+, –)
8. Verhältnis-Operatoren (<, <=, =, <>, >=, >)
9. Logische Operatoren (AND, OR, NOT)

Treten in einem Ausdruck zwei oder mehr Operatoren derselben Prioritätsstufe auf, wird der Ausdruck von links nach rechts verarbeitet. (Exponentiation wird von rechts nach links verarbeitet!) Beachten Sie, daß bei einer Operation A+B–C das Ergebnis dasselbe ist, ob Sie nun die Addition oder die Subtraktion zuerst ausführen.

Enthält ein Ausdruck ineinanderliegende Klammern, so wird die innerste Klammer zuerst bearbeitet, die äußerste zuletzt.

Für die Prioritätsstufen 3 und 4 gilt, daß die letzte Eingabe die höchste Priorität hat.

z. B.:

$-2 \wedge 4 \longrightarrow -(2^4)$

$3 \wedge -2 \longrightarrow 3^{-2}$

BEISPIEL FÜR EINE BEWERTUNGSFOLGE

Wir gehen aus von dem Ausdruck:

```
((3+5−2) * 6+2)/1Ø^LOG 1ØØ
```

Der Computer verarbeitet nun zuerst die innersten Klammern. Da "+" und "−" auf derselben Stufe stehen, wird von links nach rechts gerechnet, also die Addition zuerst ausgeführt.

```
((8−2) * 6+2)/1Ø^LOG 1ØØ
```

Dann wird die Subtraktion durchgeführt:

```
((6) * 6+2)/1Ø^LOG 1ØØ
```

Oder:

```
(6 * 6+2)/1Ø^LOG 1ØØ
```

In der nächsten Klammer wird zuerst die Multiplikation durchgeführt:

```
(36+2)/1Ø^LOG 1ØØ
```

Dann folgt die Addition:

```
(38)/1Ø^LOG 1ØØ
```

Oder:

```
38/1Ø^LOG 1ØØ
```

Nachdem nun die Klammern aufgelöst sind, hat die LOG-Funktion höchste Priorität.

```
38/1Ø^2
```

Als nächstes folgt die Exponentiation:

```
38/1ØØ
```

Und zuletzt wird die Division ausgeführt:

```
Ø.38
```

Dies ist der Wert des Ausdrucks.

ANHANG E

TASTENFUNKTIONEN

[ON BRK] (ON) wird benutzt, um den Computer anzuschalten, wenn er sich automatisch abgeschaltet hatte.

(BREAK)
Das Betätigen dieser Taste während eines Programmablaufs bewirkt eine Unterbrechung der Programmausführung.

Bei manuellen Operationen, Ein-/Ausgabe-Befehlen wie BEEP, CLOAD etc. wird mit Betätigung dieser Taste die Befehlsausführung unterbrochen.

[SHIFT] Die gelbe Taste mit der Aufschrift "SHIFT" muß benutzt werden, um Doppelfunktionen anzusprechen (die jeweils in brauner Schrift über den entsprechenden Tasten stehenden Zeichen). Z.B. wird mit [SHIFT] [?/T] das "?" angesprochen.

[C·CE] Wird benutzt, um Eingabe und Anzeige zu löschen, weiterhin hebt diese Taste eventuelle Blockaden durch Fehler auf.

[SHIFT] [CA] Löscht nicht nur den Anzeigeinhalt, sondern initialisiert darüber hinaus den Computer, d. h.
- hebt den WAIT-Timer auf;
- löscht das Anzeigeformat (USING-Format);
- hebt den TRON-Befehl auf (TROFF);
- hebt PRINT=LPRINT auf;
- hebt Blockade durch Fehler auf.

[0] bis [9] Zifferntasten

[.] Dezimalpunkt
- wird benutzt, um eine Abkürzung eines Befehls, eines Kommandos oder einer Funktion einzugeben;
- gibt in der Bestimmung eines USING-Formats die Stellung des Dezimalpunktes an.

[E] — Dient zur Eingabe des Exponenten für wissenschaftliche Schreibweise. (Dies ist die Tasto für den Buchstaben E.)

[/] Divisionszeichen.

[*] Multiplikationszeichen
- wird benutzt, um eine Feld-Variable in INPUT#, PRINT# etc. zu bestimmen.

[+] Additionszeichen

[−] Subtraktionszeichen

[SHIFT] [y^(x)] Wird benutzt, um Zahlen zu potenzieren und um das Exponenten-Anzeige-System für numerische Daten in USING-Anweisungen zu spezifizieren.

[SHIFT] [<] Werden benutzt, wenn logische Operatoren in einer IF-Sequenz
[SHIFT] [>] eingegeben werden sollen.

[DEF] Wenn eines der folgenden 18 Zeichen (A, S, D, F, G, H, J, K, L, Z, X, C, V, B, N, M, SpaCe) nach Betätigen der [DEF] -Taste gedrückt wird, wird das entsprechend gekennzeichnete Programm gestartet.

[A] bis [Z] Buchstabentasten

Diese Tasten sind Ihnen wahrscheinlich von einer gewöhnlichen Schreibmaschine her vertraut.

Auf einfachen Tastendruck erscheinen Großbuchstaben in der Anzeige. In den Kleinschriftmodus schalten Sie um, indem Sie [SML] drücken.

[SPC] Wird beim Eingeben von Zeichen oder Programmen benutzt, um einen Leerschritt zu produzieren.

[=] – In Zuweisungs-Statements: weist der links vom "=" stehenden Variablen die rechts stehenden Zeichen (Buchstabe oder Zahl) zu.

– Wird weiterhin benutzt, wenn in einer IF-Sequenz logische Operatoren eingegeben werden.

[SHIFT] [!] Nur Textzeichen

["] Kennzeichnet Textkonstante

[#] Bestimmt im USING "Format" die Anzahl der Ziffern

[$] Kennzeichnet Textvariablen

[%] Nur Textzeichen

[&] Bestimmt im USING "Format" die Anzahl der Textzeichen; kennzeichnet eine Hexadezimalzahl

[?] Zur Eingabe von CLOAD?

[SHIFT] [@] Nur Textzeichen

[SHIFT] [:] Trennzeichen zwischen mehreren Befehlen in einer Programmzeile

[SHIFT] [;] Interpunktion in Ein/Ausgabe-Anweisungen etc.

[,] Interpunktion in Ein/Ausgabe-Anweisungen etc.

[SHIFT] [(]
[SHIFT] [)] Eingabe von Klammern

[►]
- Bewegt den Cusor nach rechts (auf einmaligen Tastendruck Bewegung um eine Stelle; wird die Taste festgehalten, Dauerfunktion).
- Zur Ausführung von Playback-Anweisungen.
- Löscht bei manueller Rechnung Fehlermeldungen.

[◄]
- Bewegt den Cursor nach links (auf einmaligen Tastendruck Bewegung um eine Stelle; wird die Taste festgehalten, Dauerfunktion).
- Sonst wie unter der [►] -Taste beschrieben.

[SHIFT] [INS] Fügt einen Leerschritt ein, dabei erscheint das Zeichen "‿". Dieser Leerschritt wird vor das Zeichen gesetzt, auf dem der Cursor steht.

[SHIFT] [DEL] Löscht den Inhalt der Stelle, auf der der Cursor steht.

[SHIFT] [√‾] Zur Eingabe der Quadratwurzel

[x^2] Zur Berechnung des Quadrates.

[STAT] Zur Einstellung des STAT-Modus, wenn der RUN- oder Pro-Modus eingestellt ist.
Zur Einstellung des CAL-Modus, wenn der STAT-Modus eingestellt ist.
Mit jedem Drücken der Taste [STAT] wird zwischen STAT- und CAL-Modus umgeschaltet.

[BASIC] Zur Einstellung des RUN-Modus, wenn der CAL-Modus eingestellt ist.
Zur Einstellung des PRO-Modus, wenn der RUN-Modus eingestellt ist.
Mit jedem Drücken der Taste [BASIC] wird abwechselnd RUN und PRO eingestellt.

[SHIFT] [DRG] Zur Einstellung der Winkeleinheit.

[SHIFT] [sin] ~
[SHIFT] [e^x] Zur Eingabe der angegebenen Funktionen.

[ENTER]
- Zur Eingabe einer Programmzeile in den Compuer.
- Wird beim Schreiben von Programmen benötigt.
- Bedingt manuelle Rechnung oder direkte Ausführung eines Befehls durch den Computer.
- Zum Neustart eines Programms, das zeitweise durch einen INPUT- oder PRINT-Befehl unterbrochen wurde.

[SHIFT] [P↔NP] Stellt die Verbindung zum Drucker her bzw. unterbindet sie (sofern ein zusätzlicher Drucker an den Computer angeschlossen ist).

[SML]
- Zum Anwählen und Aufheben des Kleinschriftmodus (schaltet die Anzeige SML an bzw. aus).
- Die SML-Anzeige erscheint, wenn [SML] gedrückt wird. Wenn Sie nun die Tasten [A], [B] und [C] betätigen, werden a, b und c in der Anzeige ausgegeben. Durch erneutes Drücken von [SML] schalten Sie den Kleinschriftmodus wieder aus, die Anzeige SML verschwindet und es werden wieder Großbuchstaben in der Anzeige ausgegeben.

Die [↑] - und [↓] -Tasten haben die folgenden Funktionen, je nach angewähltem Modus und Status des Computers:

Modus	Status	[↓]	[↑]
RUN	Programmdurchführung		
	Programm vorübergehend unterbrochen,	um die nächste Zeile durchführen zu lassen.	Diese Taste festhalten, um bereits abgearbeitete Zielen zur Anzeige zu bringen.
	INPUT-, PRINT-Statement wird ausgeführt		
	Unterbrechung		
	Fehlermeldung während der Programmausführung		um die fehlerhafte Zeile zur Anzeige zu bringen
	TRON Status	zur Fehlersuche	Diese Taste festhalten, um bereits abgearbeitete Zeilen zur Anzeige zu bringen.

Wenn der Modus von RUN nach PRO geändert wird, wird dies nicht angezeigt!

PRO	Programm vorübergehend unterbrochen	um die unterbrochene Zeile anzuzeigen	wie links
	Fehler	um die fehlerhafte Zeile anzuzeigen.	wie links
	Anderer Status	um die erste Zeile anzuzeigen.	um die letzte Zeile anzuzeigen.
(Wenn die Programmzeile angezeigt wird:)			
		um die nächste Programmzeile anzuzeigen.	um die vorherige Programmzeile anzuzeigen.

- In der Anzeige ist die [ENTER] -Taste das gleiche wie ein Leerschritt.
- Wenn 12 Minuten lang keine Taste betätigt wird, schaltet sich der Computer automatisch ab.

RECHENBEREICH

Arithmetische Berechnungen:

Erster Operand
Zweiter Operand $\pm 1 \times 10^{-99}$ bis $\pm 9{,}999999999 \times 10^{99}$ und 0
Rechenergebniss

Wissenschaftliche Berechnungen:

Funktion	Rechenbereich	Anmerkung
$\sin x$ $\cos x$ $\tan x$	DEG: $\lvert x \rvert < 1 \times 10^{10}$ RAD: $\lvert x \rvert < \frac{\pi}{180} \times 10^{10}$ GRAD: $\lvert x \rvert < \frac{10}{9} \times 10^{10}$ Für tan x gelten folgende Einschränkungen DEG: $\lvert x \rvert = 90\,(2n-1)$ RAD: $\lvert x \rvert = \frac{\pi}{2}(2n-1)$ n = Ganze Zahl GRAD: $\lvert x \rvert = 100\,(2n-1)$	
$\sin^{-1} x$ $\cos^{-1} x$	$-1 \leqq x \leqq 1$	
$\tan^{-1} x$	$\lvert x \rvert < 1 \times 10^{100}$	
$\ln x$ $\log x$	$1 \times 10^{-99} \leqq x < 1 \times 10^{100}$	($\ln x = \log_e x$)
e^x	$-1 \times 10^{100} < x \leqq 230.2585092$	($e \fallingdotseq 2.718281828$)
10^x	$-1 \times 10^{100} < x < 100$	
y^x(^)	• $y > 0$: $-1 \times 10^{100} < x \log y < 100$ • $y = 0$: $x > 0$ • $y < 0$: x: ganzzahlig oder $1/x$: ungerade $-1 \times 10^{100} < x \log \lvert y \rvert < 100$	$y^x = 10^{x \cdot \log y}$
$\sqrt[x]{y}$	• $y > 0$: $-1 \times 10^{100} < \frac{1}{x} \log y < 100$, $x \neq 0$ • $y = 0$: $x > 0$ • $y < 0$: x oder $1/x$: ganzzahlig ($x \neq 0$) $-1 \times 10^{100} < \frac{1}{x} \log \lvert y \rvert < 100$	$\sqrt[x]{y} = 10^{\frac{1}{x} \cdot \log y}$
$\sqrt[3]{x}$	$\lvert x \rvert < 1 \times 10^{100}$	
$\sqrt{x}$	$0 \leqq x < 1 \times 10^{100}$	
x^2	$\lvert x \rvert < 1 \times 10^{50}$	
$\frac{1}{x}$	$\lvert x \rvert < 1 \times 10^{100}$ $x \neq 0$	
n!	$0 \leqq n \leqq 69$ (n = Ganze Zahl)	

Funktion		Rechenbereich	Anmerkung
Statistische Berechnungen	Kalkulationstabellen-Daten	$\lvert x \rvert < 1 \times 10^{50}$ $\lvert y \rvert < 1 \times 10^{50}$ $\lvert \Sigma x \rvert < 1 \times 10^{100}$ $\Sigma x^2 < 1 \times 10^{100}$ $\lvert \Sigma y \rvert < 1 \times 10^{100}$ $\Sigma y^2 < 1 \times 10^{100}$ $\lvert \Sigma xy \rvert < 1 \times 10^{100}$ $\lvert n \rvert < 1 \times 10^{100}$	
	$\bar{x}$	$n \neq 0$	
	Sx	$n \neq 1$ $0 \leqq \frac{\Sigma x^2 - n\bar{x}^2}{n-1} < 1 \times 10^{100}$	
	σx	$n \neq 0$ $0 \leqq \frac{\Sigma x^2 - n\bar{x}^2}{n} < 1 \times 10^{100}$	
	$\bar{y}$	$n \neq 0$	
	Sy	$n \neq 1$ $0 \leqq \frac{\Sigma y^2 - n\bar{y}^2}{n-1} < 1 \times 10^{100}$	
	σy	$n \neq 0$ $0 \leqq \frac{\Sigma y^2 - n\bar{y}^2}{n} < 1 \times 10^{100}$	
	r	$n \neq 0$ $0 < \lvert (\Sigma x^2 - n\bar{x}^2) \cdot (\Sigma y^2 - n\bar{y}^2) \rvert < 1 \times 10^{100}$ $\left\lvert \Sigma xy - \frac{\Sigma x \cdot \Sigma y}{n} \right\rvert < 1 \times 10^{100}$ $\left\lvert \frac{\Sigma xy - \frac{\Sigma x \cdot \Sigma y}{n}}{\sqrt{(\Sigma x^2 - n\bar{x}^2) \cdot (\Sigma y^2 - n\bar{y}^2)}} \right\rvert < 1 \times 10^{100}$	
	b	$n \neq 0$ $0 < \lvert \Sigma x^2 - n\bar{x}^2 \rvert < 1 \times 10^{100}$ $\left\lvert \Sigma xy - \frac{\Sigma x \cdot \Sigma y}{n} \right\rvert < 1 \times 10^{100}$ $\left\lvert \frac{\Sigma xy - \frac{\Sigma x \cdot \Sigma y}{n}}{\Sigma x^2 - n\bar{x}^2} \right\rvert < 1 \times 10^{100}$	
	a	a ist in derselben Bedingung wie b, und $\lvert \bar{y} - b\bar{x} \rvert < 1 \times 10^{100}$	
	y'	$\lvert a + bx \rvert < 1 \times 10^{100}$	
	x'	$\left\lvert \frac{y-a}{b} \right\rvert < 1 \times 10^{100}$	

Verteilungsfunktionen

n, n_1, und n_2: natürliche Zahl unter 10^{10}

Funktion	Rechenbereich	Anmerkung
BDS (p, n, r)	$0<p<1$ r ist natürliche Zahl kleiner oder gleich n $\frac{n!}{(n-r)!}$ oder $\frac{n!}{r!} < 10^{100}$	
FDS (n_1, n_2, x)	$0 \leqq x < 1 \times 10^{100}$ $1 \leqq n_2 \leqq 999$ wenn $1 \leqq n_1 \leqq 39$ $1 \leqq n_1 \leqq 999$ wenn $1 \leqq n_2 \leqq 39$	n_1, n_2 sind begrenzt
NDS (x)	$0 \leqq \lvert x \rvert < 1 \times 10^{100}$	
PDS (m, x)	$0 \leqq x \leqq 69$ (x: ganze Zahl) $m>0$ wenn $x=0$ $0 \leqq m < 10^{\frac{100}{x}}$ wenn $x \geqq 1$	
PFD (n_1, n_2, p)	$0.001 \leqq p \leqq 1.000$ (Inkrement: 0.001) $1 \leqq n_2 \leqq 999$ wenn $1 \leqq n_1 \leqq 39$ $1 \leqq n_1 \leqq 999$ wenn $1 \leqq n_2 \leqq 39$	n_1, n_2 sind begrenzt
PND (p)	$0.001 \leqq p \leqq 0.999$ (Inkrement: 0.001)	
PTD (n, p)	$0.001 \leqq p \leqq 1.000$ (Inkrement: 0.001)	
PXD (n, p)	$0.001 \leqq p \leqq 0.999$ (Inkrement: 0.001)	
TDS (n, x)	$0 \leqq \lvert x \rvert < 1 \times 10^{100}$	
XDS (n, x)	$0 \leqq \lvert x \rvert < 1 \times 10^{100}$	

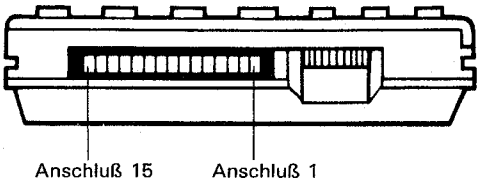
Die Genauigkeit beträgt in der Regel für die Fließkomma-Schreibweise ±1 in der 10. Stelle und für die wissenschaftliche Schreibweise ±1 in der 9. Nachkommastelle der Mantisse.
Die Berechnungsgenauigkeit sinkt jedoch in der Nähe des Singulär-Punktes und des Umkehrpunktes der Funktion ab.

Ferner ist zu beachten, daß bei Kettenrechnungen eine Kumulierung des Fehlers erfolgt und damit mit jedem Rechengang eine Verschlechterung der Genauigkeit einhergeht. Der gleiche Effekt tritt rechnerintern bei der Durchführung von Funktionen wie y^x und $\sqrt[x]{y}$ auf.

ANHANG F

SIGNALE DES E/A-ANSCHLUSSES

Der Computer ist mit einem 15poligen Anschluß für ein E/A-Terminal ausgestattet. Die Anschlüsse und ihre Symbole werden anschließend beschrieben.



Anschluß	Name	Signal	E/A	Funktion
1	Frame Ground	FG		Massenanschluß für Wartung
2	Transmit Data	SD	A	DC Ausgabesignal
3	Receive Data	RD	E	DC Eingabesignal
4	Request Data	RS	A	HIGH: Sendesignal
5	Clear To Send	CS	E	HIGH: Übertragung
7	Signal Ground	SG	—	Masse für alle Signale
8	Data Carrier Detect	CD	E	HIGH: Trägersignal erhalten
10		VC		Energieversorgung
11	Receive Ready	RR	A	HIGH: Empfang möglich
12	Peripheral acknowledge	PAK	E	Siehe Hinweis 3
13		VC		Energieversorgung
14	Data Terminal Ready	ER	A	HIGH: Terminal bereit
15	Periphenal request	PRQ	A	Siehe Hinweis 3

Hinweis:

1. HIGH (hoch) ist der VC-Spannungspegel, LOW (niedrig) ist SG-Spannungspegel.
2. Der **PC-1425** verwendet CMOS-Bauteile. Die Anwendung von Spannungen, die den zulässigen Bereich zwischen SG und VC überschreiten, kann den Computer beschädigen.
3. Dieses Signal bestätigt, ob das angeschlossene Gerät der Drucker CE-140P ist oder nicht. Dieses Signal wird nicht für die Kommunikation mit anderen Geräten verwendet.

ANHANG G

TECHNISCHE DATEN

Modell:	**PC-1425** Taschencomputer
Prozessor:	8 Bit CMOS CPU
Programmiersprache:	BASIC
Speicherkapazität:	System-ROM 136K Bytes RAM: System: ca. 1,1K Bytes Anwender: Fester Speicherbereich 208 Bytes (A–Z, A$–Z$) Programm-/Datenbereich 6878 Bytes (bei Verwendung der CE-212M)
Stack:	Unterroutine: 10 Stacks Funktion: 16 Stacks FOR-NEXT: 5 Stacks Daten: 8 Stacks
Operatoren:	Addition, Subtraktion, Multiplikation, Division, trigonometrische und inverstrigonometrische Funktionen, logarithmische und Exponentialfunktionen, Verteilungsfunktionen, Quadrat und Quadratwurzel, Vorzeichen, Absolut, Integral, Verhältnis-Operatoren, logische Operatoren.
Rechengenauigkeit:	10 Stellen (Mantisse) + 2 Stellen (Exponent)
Editiermöglichkeit:	Cursor rechts und links, Zeile auf- und abwärts, Zeicheneinfügung, Zeichenlöschung
Speicherschutz:	CMOS, batteriegestützt
Serieller E/A-Anschluß und seine Möglichkeiten:	
Standards:	Start-Stop-Übertragungssystem (asynchron). Nur Halb-Duplex.
Baud-Raten:	300, 600, 1200 Baud
Datenbits:	7 oder 8 Bits
Paritätsbits:	Gerade, ungerade oder keine Parität
Stopbit:	1 oder 2 Bits

Anschlüsse:	15poliger Anschluß (für externe Geräte)
Ausgabesignal:	C-MOS Niveau (4—6 Volt)
Interface-Signale:	Eingaben: RD, CS, CD, PAK Ausgaben: SD, RS, RR, ER, PRQ Andere: SG, FG, VC
Anzeige:	24-Stellen-Flüssigkristallanzeige mit 5 x 7-Punkt-Zeichen
Tastatur:	78 Tasten, alphabetisch, numerisch, Sonderzeichen, Funktionen, numerische Schablone, definierbare Tastenfunktionen
Stromversorgung:	6,0V DC-Lithium-Zellen Typ: CR-2032 x 2
Stromverbrauch:	6,0V DC ca. 0,03 W Ca. 120 Stunden bei Dauerbetrieb unter normalen Bedingungen. (Gilt bei einer Betriebstemperatur von 20°C, wenn sich eine Betriebsstunde aus 10 Minuten für Bedienung und Programmausführung und 50 Minuten Anzeigezeit zusammensetzt.)
	* Abhängig von Arbeitsmethoden und verwendetem Batterietyp schwankt die Betriebszeit leicht.
Betriebstemperatur:	0 Grad Celsius bis 40 Grad Celsius
Abmessungen:	182 (B) x 72 (T) x 16 (H) mm
Gewicht:	ca. 175 g (mit Batterien und RAM-Karte)
Zubehör:	Deckel, 2 eingebaute Lithium-Zellen, Tastaturschablone, Handbuch und RAM-Karte (CE-212M)
Zusatzgeräte:	RAM-Karten – 8K Bytes RAM (CE-201M) – 16K Bytes RAM (CE-202M) – 32K Bytes RAM (CE-203M) – 2K Bytes RAM (CE-210M) – 4K Bytes RAM (CE-211M) – 8K Bytes RAM (CE-212M) – 16K Bytes RAM (CE-2H16M) Cassettenrecorder (CE-152) Drucker/Cassetten-Interface (CE-126P) Druckers (CE-140P, CE-515P) etc.

ANHANG H

DIE BENUTZUNG VON PROGRAMMEN, DIE FÜR ANDERE PC-MODELLE ENTWICKELT WURDEN

Programme, die für die folgenden Computer geschrieben wurden, können mit geringfügigen Veränderungen auf dem PC-1425 verwendet werden.

PC-1210 Serie: PC-1210/11
PC-1245 Serie: PC-1245/46/47
PC-1250 Serie: PC-1250/51
PC-1260 Serie: PC-1260/61
PC-1350 Serie: PC-1350
PC-1401 Serie: PC-1401/02
PC-1403 Serie: PC-1403
PC-1450 Serie: PC-1450
PC-1460 Serie: PC-1460
PC-2500 Serie: PC-2500

Obwohl sich die Funktionen der obigen Modelle geringfügig unterscheiden, können die mit diesen Modellen erstellten Programme auf dem PC-1425 verwendet werden, wenn die nachstehend beschriebenen Modifikationen durchgeführt werden.

Hinweis 1: Der PC-1425 kann Programme von Cassette lesen, die mit Computern der Serien PC-1210, PC-1245 und PC-1250 auf Cassette geschrieben wurden, aber umgekehrt können Programme, die mit dem PC-1425 erstellt und auf Cassette geschrieben wurden, nicht mit den Computern der drei obengenannten Serien gelesen werden.

Hinweis 2: Programm-Cassetten für die Serien PC-1245 und PC-1210, auf die mehrere Programme mit dem MERGE-Befehl aufgenommen wurden, können nicht mit dem PC-1425 verwendet werden. Vielmehr müssen die Programme einzeln mit dem MERGE-Befehl in den PC-1425 geladen werden.

Hinweis 3: Programme, die mit der Serie PC-1250 erstellt werden und die Befehle POKE oder CALL enthalten, können nicht auf dem PC-1425 ausgeführt werden. Wird die Ausführung auf dem PC-1425 versucht, kann es vorkommen, daß der Computer keine Eingabe von den Tasten annimmt.

MODIFIKATIONEN VON PROGRAMMEN DER PC-1245 SERIE (PC-1250 SERIE)

Wenn Sie die Programme, die für die PC-1245 Serie entwickelt wurden, auf dem PC-1425 benutzen wollen, müssen Sie folgende Änderungen vornehmen:

1. Multiplikation ohne das " * "-Zeichen
Bei der PC-1245 Serie kann der Operator für die Multiplikation (*) fortgelassen werden (z.B. AB anstatt A * B, DC statt D * C). Dies ist auf dem PC-1425 nicht möglich, da dieser zwei aufeinanderfolgende Zeichen als einfache Variable behandelt. Benutzen Sie daher die Schreibweise rechts im Beispiel:

(z.B.) A = SIN BC ⟶ A = SIN (B * C)

2. Definition von Index-Variablen mit dem DIM-Befehl
Wenn auf den Computern der PC-1245 Serie der Befehl DIM A(3Ø) ausgeführt wird, werden Speicherplätze von A(27) bis A(3Ø) als Erweiterung eines Definitionsbereiches einer festen Variablen reserviert.

Auf dem PC-1425 wird jedoch bei der Ausführung von DIM A(3Ø) ein getrennter Speicherbereich für die Feld-Variablen A(Ø) bis A(3Ø) und das Feld mit dem Namen A reserviert.

Wenn Sie Index-Variable als Erweiterung von Festvariablen definieren wollen, benutzen Sie bitte die rechts im Beispiel angegebene Schreibweise:

(z.B.) DIM A(3Ø) ⟶ A(3Ø) = Ø

3. Datenein-/ausgabefehl für Dateien auf Cassette:
Bei der PC-1245 Serie wird mit dem Befehl PRINT#C der Inhalt der Variablen C und aller folgenden abgespeichert.

Beim PC-1425 jedoch wird mit diesem Befehl lediglich der Inhalt der Variablen C auf Band gespeichert. Um den Inhalt einer bestimmten Variablen und aller ihr folgenden auf Band abzuspeichern, benutzen Sie bitte die Schreibweise rechts im Beispiel:

(z.B.) PRINT#A ⟶ PRINT#A*
INPUT#C ⟶ INPUT#C*

4. Wert einer Schleifenvariablen nach Beendigung einer FOR-NEXT-Schleife

Der Wert einer Schleifenvariablen, den Sie nach Ausführung einer FOR-NEXT-Schleife erhalten, unterscheidet sich beim PC-1425 von dem bei der PC-1245 Serie. Wird der Wert einer Schleifenvariablen in einem Bedingungsausdruck benutzt, müssen Sie ihn beim PC-1425 gegenüber der PC-1245 Serie um 1 erhöhen.

(z.B.)
```
1Ø  FOR I=Ø TO 1Ø
5Ø  NEXT I
6Ø  IF I=1Ø THEN 1ØØ
```

Ändern Sie den Wert in Zeile 6Ø folgendermaßen:

```
6Ø  IF  I=11  THEN  11Ø                    PC-1425
```

(Auf dem PC-1425 müssen Sie den Wert einer Schleifenvariablen um eins erhöhen. Die Anzahl der Schleifendurchläufe bleibt jedoch gleich.)

5. Umdefinition von =
Die Gleichheitstaste = funktioniert beim PC-1460 nicht als definable Key. Daher muß in Programmen, in denen die Gleichheitstaste definiert ist, eine andere Taste verwendet werden.

(z.B.) 100 "=": . . . → 100 "N" : . . .

6. Exponentialzeichen "IE"
Der PC-1425 benützt den Großbuchstaben "E" für die Darstellung der Exponentiation. Die folgenden Änderungen werden dadurch erforderlich:

A = 1.234 IE 5 ⟶ A = 1.234E5
B = IE 6 ⟶ B = 1E6

Wird ein Programm der PC-1245 Serie von Band in den PC-1425 eingelesen, wird diese Veränderung automatisch vom PC-1425 durchgeführt.)

7. Der Zeichen-Code der PC-1245 Serie unterscheidet sich geringfügig von dem des PC-1425. Wenn die folgenden Codes durch die CHR$-Funktion zugewiesen werden, ändern Sie die Codes:

ASCII	PC-1245	PC-1425
39 (&27)	‾	,
91 (&5B)	√	[
92 (&5C)	¥	\
93 (&5D)	π	]
96 (&60)	IE	‘
250 (&FA)	– (Error)	‾
251 (&FB)	– (Error)	π
252 (&FC)	– (Error)	√

Anmerkung: Wie oben gezeigt, verfügt der PC-1425 nicht über das Zeichen IE.

ZUSÄTZLICHE MODIFIKATIONEN

1. Die Modelle der PC-1245 Serie verfügen über eine Programmzeilenspanne von 1 – 999. Der PC-1425 hat eine erweiterte Spanne von 1 – 65279. Aus diesem Grund beansprucht die Zeilennummer 3 Bytes im RAM (bei der PC-1245 Serie nur 2 Bytes.). Die Änderung wird automatisch ausgeführt, wenn Sie Programme von Cassette in den Computer einladen. Es besteht die Möglichkeit eines Speicherüberlaufs, wenn Sie zu lange Programme laden oder ausführen lassen (ERROR 6). Wenn eine Zeile etwa 8Ø Bytes umfaßt, kann durch diese Veränderung das Zeilenende gelöscht werden.

2. Wenn Sie ein Programm der PC-1245 Serie in den PC-1425 einladen, wird aufgrund der Änderung in der Zeilennummerierung (von 2 auf 3 Bytes) die BUSY-Anzeige noch für 1 oder 2 Sekunden aufleuchten, nachdem das Band zu laufen aufgehört hat. Während dieser Zeit wird das "*"-Zeichen in der rechten unteren Ecke des Zeilendisplays sichtbar sein.

ÄNDERUNGEN BEZÜGLICH DER PC-1210 SERIE

Um Programm der PC-1210 Serie auf dem PC-1425 zu benutzen, müssen diese in gleicher Weise geändert werden, wie Programme der PC-1245 Serie (außer den Punkten 2 und 7). Zusätzlich sind folgende Änderungen notwendig:

1. IF-Statement

Beispiel:

Die folgende Programmzeile eines Programms der PC-1210 Serie

```
5Ø IF A>L PRINT "A"
```

wird vom PC-1425 interpretiert als

```
5Ø IF A> LPRINT "A"
```

und erzeugt eine Fehlermeldung.

Der Fehler taucht auf, weil ein Befehl (LPRINT) nicht in der PC-1210 Serie verwendet wird, wohl aber beim PC-1425.

Um dieses Problem zu lösen, sollten Sie in das IF-Statement einen THEN-Befehl einfügen:

```
5Ø IF A>L THEN PRINT "A"
```

2. USING-Format

Der USING-Befehl unterscheidet sich bei der PC-1210 Serie und dem PC-1425. in folgender Weise:

Beispiel:

```
1Ø A = −123.456
2Ø PAUSE USING "####.##"; A
3Ø PAUSE A, USING "####"; A
```

Bei der Ausführung dieses Programms wird folgendes angezeigt:

* PC-1210		−123.45
	−123	−123
* PC-1425		
		−123.45
	−123.45	−123

In der PC-1210 Serie wird die Anzeige auf der linken Seite im gleichen Format angezeigt wie auf der rechten Seite. Beim PC-1425 gestaltet sich die Anzeige entsprechend dem vorangegegangenen Format-Befehl.

Dies bezieht sich nicht nur auf den PAUSE-Befehl, sondern gilt auch für PRINT- und LPRINT-Befehle.

3. Auslassen von ")"

In der PC-1210 Serie kann das ")" weggelassen werden, sofern es direkt vor [ENTER] oder ":" vorkommt. Beim PC-1425 kann das ")"-Zeichen nicht ausgelassen werden.

Deshalb müssen Sie es einfügen, wenn Sie Programme übernehmen wollen.

4. PRINT-Befehl

Der PC-1425 hat einen PRINT-Befehl für die Anzeige und einen LPRINT-Befehl für die Ausgabe auf dem Drucker. Alle PRINT-Befehle können umgeändert werden in LPRINT-Befehle durch ein PRINT=LPRINT-Statement.

Die PC-1210 Serie hat keinen LPRINT-Befehl. Um bei Übernahme von Programmen auf dem PC-1425 Ausgaben auf dem Drucker zu erhalten, müssen Sie ein PRINT=LPRINT-Statement in das Programm einfügen.

5. Variablen

Wird ein Programm mit RUN auf der PC-1210 Serie gestartet, werden alle Variablen beibehalten. Beim PC-1425 werden alle Variablen von A(27) an gelöscht.

Deshalb ist es nötig, falls die Variablen beibehalten werden sollen, das Programm auf dem PC-1425 mittels GOTO oder der [DEF] -Taste zu starten.

MODIFIKATIONEN FÜR DIE PC-1260 SERIE

(1) Zeichencode-Modifikation
Der Zeichencode 96 (&60) ist in der PC-1260 Serie ein Leerzeichen, aber im PC-1425 ein Hochkomma ('). Wenn daher in einem Programm der CHR$-Befehl mit dem Zeichencode 96 zur Spezifizierung einer Leerstelle vorkommt, muß dieser Zeichencode in 32 (&20) verändert werden.

(2) Befehle CLS, CURSOR
Der PC-1425 verfügt nicht über die Befehle CLS und CURSOR. Daher müssen diese Befehle in Programmen, die auf dem PC-1425 laufen sollen, gelöscht und verändert werden.

MODIFIKATIONEN FÜR DIE PC-1401, PC-1403, PC-1450 UND PC-1460 SERIE

Programme, die auf Computern der Serie PC-1401, PC-1403, PC-1450 und PC-1460 erstellt wurden, können ohne Veränderungen auf dem PC-1425 laufen.

MODIFIKATIONEN FÜR DIE SERIEN PC-1350 UND PC-2500

Der PC-1425 verfügt nicht über die folgenden Befehle, so daß diese verändert werden müssen, wenn sie in einem Programm vorkommen.

CLS, CURSOR, MEM$, GCURSOR, GPRINT, LINE, POINT, PRESET, PSET, (TEST)

Programmierungsbeispiele

Nach Lesen der Beschreibung zu den verschiedenen Funktionene in den vorangegangenen Kapiteln, haben Sie sich mit einer Anzahl von Programmbefehlen vertraut gemacht. Zur Zusammenstellung von Befehlen für Applikationsprogramme in BASIC, ist es jedoch unbedingt notwendig, zusätzlich zu den in dieser Anleitung erläuterten Programmen auch eigene praktische Applikationsprogramme zu schreiben.

Genauso wie sich Autofahren nur durch die Handhabung des Lenkrads oder Tennisspielen durch ständiges Schwingen des Schlägers erlernen läßt, kann auch effizientes Programmieren nur durch Üben möglichst vieler Programme gemeistert werden, ganz gleich wie gut Sie jede der Übungen beherrschen.

Es ist somit äußerst wichtig für Sie, zunächst Programme zu üben, die Sie nicht selbst geschrieben haben. In diesem Kapitel werden einige Programmierbeispiele mit Anwendung verschiedener Befehle in BASIC als Übungs- und Bezugsmaterial vorgestellt.

Zum besseren Verständnis der Programmbeispiele in diesem Kapitel werden nachfolgend die in diesen Beispielen verwendeten Bezeichnungen erläutert:

(1) PROGRAMMLISTUNG
Alle Programmlisten in den Programmierbeispielen werden mit der Hardcopy des Druckers CE-126P oder CE-140P in tatsächlicher Größe oder einem um 55% verkleinerten Format ausgegeben.

(2) PROGRAMMKAPAZITÄT
Am Ende einer jeden Programmliste wird die Kapazität des Programms selbst in der Anzahl von Bytes angebeben.

(3) AUSDRUCKEN
Bei auszudruckenden Programmen wird die Ausgabe für das ausgeführte Programm mit dem Drucker CE-126P oder CE-140P in tatsächlicher Größe oder einem um 55% verkleinerten Format ausgegeben. Andere verkleinerte Ausdrucke als das 55%-Format werden zusammen mit der entsprechenden Verkleinerungsrate angegeben.

(4) SPEICHERINHALT
In der Speicherinhalttabelle jedes Programmbeispiels werden verschiedene Variablen mit vorherbestimmter Anwendung (z.B. solche, die zur Erhaltung von sofortigen Kalkulationszwischenergebnissen in den Arbeitsbereich einzuspeichern sind usw.) mit dem Prüfhäkchen "√" angezeigt.

(5) Bei Verwendung von CE-515P (oder CE-516P)
In der Inhaltstabelle dieses Kapitels zeigt "P" vor einem Programmtitel an, daß der CE-140P, CE-515P oder CE-516P als Peripheriegerät für den Programmablauf eingesetzt werden kann. Bei Verwendung von CE-515P oder CE-516P sind jedoch folgende Punkte zu beachten:

1. Einstellung des DIP-Schalters
* Mit CE-515P..... Alle DIP-Schalterstifte auf "OFF" (Aus) stellen.
* Mit CE-516P..... Die mit 1 bis 5 nummerierten Schalterstifte auf "OFF" (Aus) stellen und Schalterstift Nr. 6 auf "ON" (Ein).

2. Zu verwendendes Papier
In jedem Fall eine Papierrolle verwenden.

Achtung:

Bei Verwendung des CE-515P oder CE-516P unterscheiden sich die Hardcopy-Ausgaben teilweise in folgenden Punkten von jenen des CE-140P:

(1) Farbe
CE-140P: violett → CE-515P (oder CE-516P): blau
(2) Zeichen
CE-140P: *a* → CE-515P: *a*
(CE-516P druckt die gleichen Zeichen wie CE-140P.)

SHARP CORPORATION und/oder ihre Zweigniederlassungen tragen keine Verantwortung oder Haftung für etwaige finanzielle Verluste oder Schäden, die durch die Anwendung irgendeines der in dieser Anleitung beschriebenen Programmbeispiele entstehen. Bei der Anwendung dieser Programme ist daran zu denken, daß diese Programme ihren Anforderungen mitunter nicht vollständig gerecht werden oder manche Programm nicht so präzise sind, wie Sie es eventuell erwarten. Die Daten eines jeden Programmbeispiels sind deshalb sorgfältig zu überprüfen, um zu sehen, ob diese den Ansprüchen genügen. Sollte dies nicht der Fall sein, können diese durch Änderungen den Applikationszwecken angepaßt werden.

INHALT

Programmtitel Seite

- Umwandlunge zwischen orthogonalen Koordinaten und Polarkoordinaten (dreidimensional) 288
P • Exponentielle Regression und Plotten 293
- Vergleich von Mittelwerten (Verbindung der Kalkulationstabellen-Daten mit BASIC) 300
- Berechnung des Flächeninhalts eines Vielecks 304
P • Erstellung einer Kurvengrafik 311
- Übertragung von Programmdateien 320
- Zahlenspiel 325

PROGRAMMTITEL: **Umwandlungen zwischen othogonalen Koordinaten und Polarkoordinaten (dreidimensional)**

Dies ist ein sehr nützliches Programm zur Durchführung von Umwandlungen zwischen orthogonalen Koordinaten und Polarkoordinaten (sphärische Koordinaten) in drei Dimensionen. Nach Eingabe aller Daten für die Umwandlung wird das Resultat der Umwandlung entsprechend der zu dieser Zeit gültigen Winkeldefinition erzielt.

■ BEDIENUNG

1. [DEF] [A] drücken (zur Umwandlung von orthogonalen Koordinaten in Polarkoordinaten).
   Wird der Wert jedes ortogonalen Koordinaten, x, y und z entsprechend der Abfrage auf dem Display eingegeben, erscheint der Wert jedes Polarkoordinaten r, θ und φ in der angegebenen Reihenfolge, und das Programm ist beendet.

2. [DEF] [B] drücken (zur Umwandlung von Polarkoordinaten in orthogonale Koordinaten).
   Wird der Wert jedes Polarkoordinaten R, θ und φ entsprechend der Abfrage auf dem Display eingegeben, erscheint der Wert jedes orthogonaen Koordinaten x, y und z in der angegebenen Reihenfolge, und das Programm ist beendet.

Achtung: Ist die Einheit der Winkeldefinition DEG, ergibt dies ein Umwandlungsresultat in Gradeinheiten. Ebenso ergibt sich ein Umwandlungresultat in Radianten-Einheiten, wenn die Winkeldefinition als RAD angegeben wird.

■ ACHTUNG

1. Umwandlungen von orthogonalen Koordinaten in Polarkoordinaten $r=0$ und θ unbegrenzt, wenn $r=y=0$

   $r = \sqrt{x^2+y^2+z^2}$

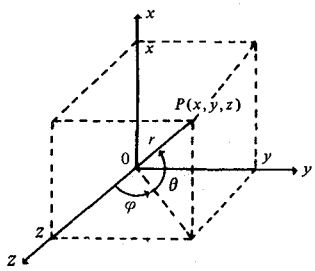
   $\theta=\sin^{-1}(z/r)$
   $\varphi=\tan^{-1}(y/x)$ wenn $x>0$
   $\varphi=90°$ wenn $x=0$ und $y\geqq 0$
   $\varphi=-90°$ wenn $x=0$ und $y<0$
   $\varphi=\tan^{-1}(y/x)+180°$ wenn $x<0$ und $y\geqq 0$
   $\varphi=\tan^{-1}(y/x)-180°$ wenn $x<0$ und $y<0$

2. Umwandlungen von Polarkoordinaten in orthogonale Koordinaten

$x = r \cos\theta \cdot \cos\varphi$

$y = r \cos\theta \cdot \sin\varphi$

$z = r \sin\theta$



■ BEISPIELE

1. Wandeln Sie orthogonale Koordinaten in Polarkoordinaten um.

   $x = -1$

   $y = 2$

   $z = -3$

   Winkeldefinitionsangabe: DEG

2. Wandeln Sie Polarkoordinaten in orthogonale Koordinaten um.

   $r = 3.741657387$

   $\theta = -53.3077479$

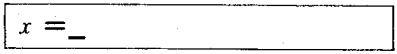
   $\varphi = 116.5650512$

   Winkeldefinitionsangabe: DEG

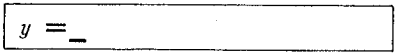
■ TASTENBETÄTIGUNG

< Orthogonale Koordinaten → Polarkoordinaten >

1. DEF A



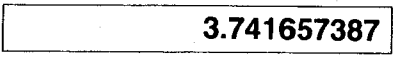
2. −1 ENTER



3. 2 ENTER

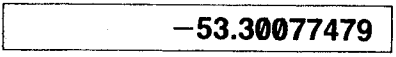
4. −3 ENTER

5. ENTER

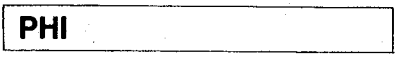


6. ENTER

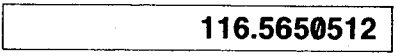
7. ENTER



8. ENTER



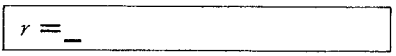
9. ENTER



10. ENTER

< Polarkoordinaten → Orthogonale Koordinaten >

1. DEF B

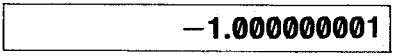


2. 3.741657387 ENTER

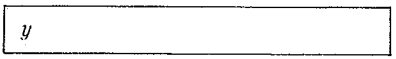
3. −53.30077479 ENTER

4. 116.5650512 ENTER

5. ENTER



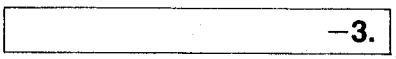
6. ENTER



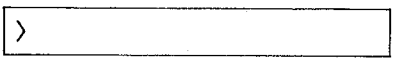
7. ENTER

8. ENTER

9. ENTER



10. ENTER



■ PROGRAMMLISTING

```
 10:"A":INPUT "x=";X
 20:INPUT "y=";Y
 30:INPUT "z=";Z
 40:R=SQR (X*X+Y*Y+Z*Z):
    IF R=0 PAUSE "r=0 Un
    defined":END
 50:C=ASN (Z/R):IF X>0
    LET F=ATN (Y/X):GOTO
    80
 60:A=(Y=0)+SGN Y:IF X=0
    LET F=A*ACS 0:GOTO 8
    0
 70:F=ATN (Y/X)+A*ACS -1
 80:WAIT :PRINT "r":
    PRINT R:PRINT "THETA
    ":PRINT C:PRINT "PHI
    ":PRINT F:END
 90:"B":INPUT "r=";R
100:INPUT "THETA=";C
110:INPUT "PHI=";F
120:Y=R*COS C:X=Y*COS F:
    Y=Y*SIN F:Z=R*SIN C
130:WAIT :PRINT "x":
    PRINT X:PRINT "y":
    PRINT Y:PRINT "z":
    PRINT Z:END
```

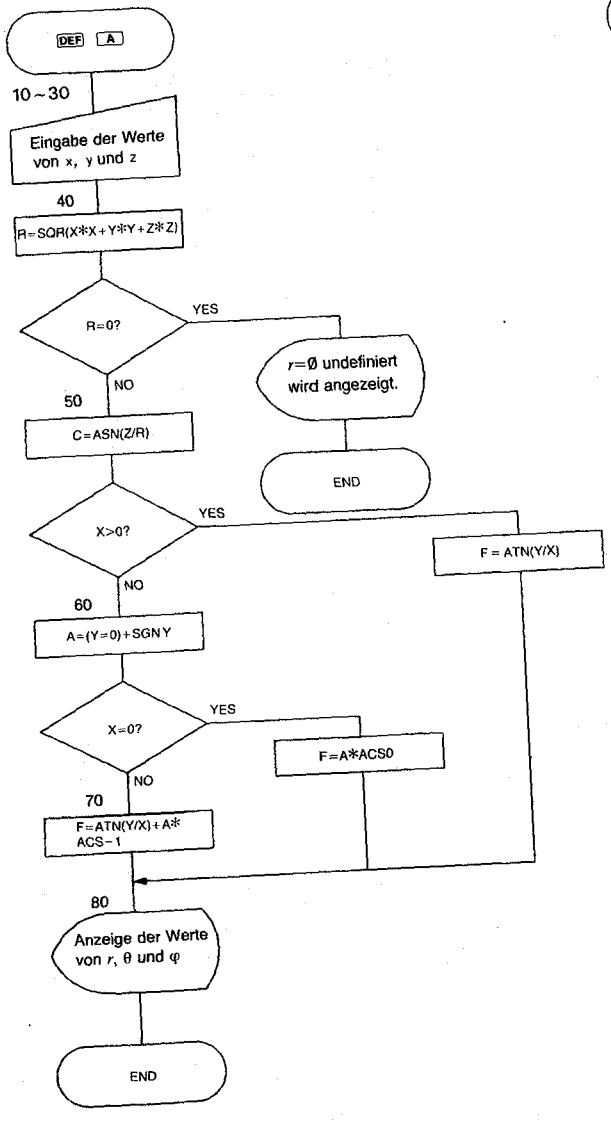
300 Bytes

■ SPEICHERINHALT

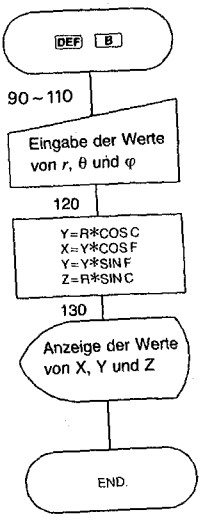
A	$\sqrt{}$
C	θ
F	φ
R	r
X	x-Koordinate
Y	y-Koordinate
Z	z-Koordinate

■ ABLAUFDIAGRAMM

(Orthogonale Koordinaten → Polarkoordinaten)



(Polarkoordinaten → Orthogonale Koordinaten)



PROGRAMMTITEL: **Exponentielle Regression und Plotten**

Erforderliche Peripheriegeräte CE-140P oder CE-515P (oder CE-516P) und CE-516L

Im Laufe der Natur treten im Zuge exponentieller Funktionen zahlreiche Veränderungen ein, wie sich z.B. im Verhältnis zwischen Zeit und Temperatur (wenn etwas heißes kalt wird) oder dem Verhältnis zwischen Zeit und der Spaltung des Atomkerns usw. zeigt. Dieses Programm löst die Werte von Koeffizienten a und b und den Koeffizient von Korrelationskoeffizient r durch Anlegen von Dateneingaben x und y an die Exponentialkurve $y=a \cdot b^x$. Die Exponentialkurve wird dann vom Drucker mit den Eingabedaten und geschätzten Werten in der Grafik ausgegeben.

■ BEDIENUNG

1. DEF A drücken. (Programmstart)
2. Gemäß der Abfrage auf dem Display die Nummer der Daten (N) eingeben. Werden die Daten x und y, wie angegeben von der Nummer der Daten (N), wiederholt eingeben, werden die Koeffizienten a und b und der Koeffizient von Korrelation r berechnet und durch den Drucker ausgedruckt.
3. Durch Drücken von DEF B , wird die Exponentialkurve in grün ausgedruckt und x und y der Eingabedaten in der Grafik in violett aufgezeichnet.
4. Anschließend werden, durch Eingeben der geschätzten x-Werte die Eingabedaten in rot in der Grafik aufgezeichnet und die geschätzten x- und y-Werte ausgedruckt. Das Programm ist damit beendet.

Achtung: Die Anzahl der Eingabedaten muß innerhalb folgender Begrenzung liegen:
CE-212-M (oder CE-201M): etwa 180
CE-2H16M (oder CE-202M): 225

■ ACHTUNG

Eine unendliche (N) Anzahl an gegebenen Punkten $(X_1, Y_1), (X_2, Y_2) \ldots (X_n, Y_n)$ wird an Exponentialfunktion $y=a \cdot b^x$ angelegt. Diese Funktion wird durch $Y=l_n y$, $A=l_n a$, $B=l_n b$ ersetzt und in lineare Ausdrücke $Y=A+Bx$ umgewandelt. Nach der Methode des kleinsten Fehlerquadrats werden A und B anschließend wie folgt bestimmt:

$$B=\frac{\sum x_i \cdot Y_i - n \cdot \bar{x} \cdot \overline{Y}}{\sum x_i^2 - n \cdot \bar{x}^2}, \quad A=\overline{Y}-B \cdot \bar{x}$$

mit: $\quad Y_i=l_n y_i, \quad \overline{Y}=\frac{1}{n}\sum_{i=1}^{n} Y_i, \quad \bar{x}=\frac{1}{n}\sum_{i=1}^{n} x_i$

Sind A und B einmal bestimmt, können a und b aus $a=e^A$, $b=e^B$ erlangt werden und ebenso auch der Koeffizient von Korrelation r zwischen x und y.

■ BEISPIEL

x	0.5	1.2	3.1	7.4
y	7.01	11.72	44.54	936.71

N=4

* Die obigen Daten an $y=a \cdot b^x$ anlegen und die y-Werte mit den geschätzten x-Werten, angenommen als jeweils 2, 4, 6 und 6.5 schätzen.

■ Bei Verwendung von CE-515P oder CE-516P

Zeilen 130 und 150 in der Programmliste auf Seite 292 ändern und wie folgt Zeile 375 zur Liste addieren:

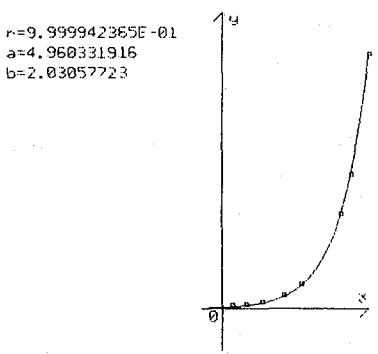
```
130:GOSUB 375

150:"B":M=F*300: GOSUB 375


375:CLOSE : OPEN "1200,N,8,1,A,L
```

■ AUSDRUCKEN

(Ausgedruckt in Farbe. Siehe Seite I)



■ TASTENBETÄTIGUNG

< Dateneingabe >

1. DEF A

N=_

2. 4 ENTER

x (1) =

?

3. 0.5 ENTER

y (1) =

?

4. 7.01 ENTER

x (2) =

?

5. 1.2 ENTER

y (2) =

?

6. 11.72 ENTER

x (3) =

?

⋮

Daten auf die gleiche Weise wie oben eingeben.

⋮

7. 936.71 ENTER

>

<Ausgabe des Resultats>

1. DEF B

Est. x=_

2. 2 ENTER

Est. x=_

3. 4 ENTER

Est. x=_

4. 6 ENTER

Est. x=_

5. 6.5 ENTER

>

■ PROGRAMMLISTING

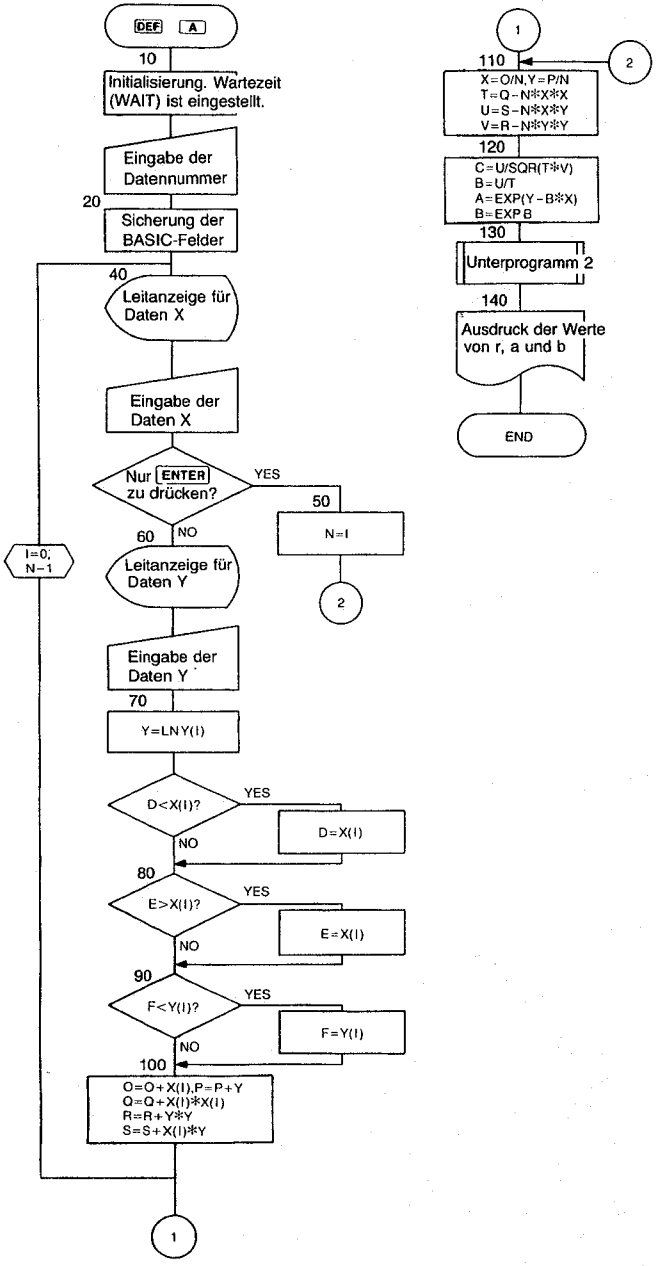
```
10:"A": CLEAR : WAIT 0: INPUT "N=";N
20:DIM X(N-1),Y(N-1),XS(N-1),YS(N-1):E
   =10^8:D=-E:F=D
30:FOR I=0 TO N-1
40:PAUSE "x("+STR$ (I+1)+")=": INPUT X
   (I): GOTO 60
50:N=I: GOTO 110
60:PAUSE "y("+STR$ (I+1)+")=": INPUT Y
   (I)
70:Y=LN Y(I): IF D<X(I) LET D=X(I)
80:IF E>X(I) LET E=X(I)
90:IF F<Y(I) LET F=Y(I)
100:O=O+X(I):P=P+Y:Q=Q+X(I)*X(I):R=R+Y*
   Y:S=S+X(I)*Y: NEXT I
110:X=O/N:Y=P/N:T=Q-N*X*X:U=S-N*X*Y:V=R
   -N*Y*Y
120:C=U/SQR (T*V):B=U/T:A=EXP (Y-B*X):B
   =EXP B
130:GOSUB 380
140:LPRINT "r=";C: LPRINT "a=";A:
   LPRINT "b=";B: END
150:"B":M=F/300: GOSUB 380
160:IF E>=0 LET Z=25:L=D/175: GOTO 180
170:L=(D+ABS E)/200:Z=ABS E/L+5
180:GRAPH : GLCURSOR (Z+230,-225)
185:SORGN
190:LLINE (-Z,0)-(200-Z,0)-(200-Z-10,-1
   0),0
200:LLINE (200-Z,0)-(200-Z-10,10):
   LPRINT "Px"
210:LLINE (0,-50)-(0,350)-(-10,340)-(0,
   350)
215:LLINE -(10,340): LPRINT "Py"
220:GLCURSOR (-15,-15): LPRINT "P0"
230:COLOR 1: FOR I=0 TO N-1:J=X(I)/L:K=
   Y(I)/M: GOSUB 370: NEXT I
240:COLOR 2:J=-Z
245:GLCURSOR (J,A*B^(J*L)/M)
250:J=J+2: IF J>200-Z GOTO 270
260:K=A*B^(J*L)/M: IF K>350 THEN 270
265:LLINE -(J,K): GOTO 250
270:I=0: COLOR 3
280:INPUT "Est. x=";XS(I): GOTO 300
290:GOTO 320
300:J=XS(I)/L:YS(I)=A*B^XS(I):K=YS(I)/M
   : IF K<=350 GOSUB 370
310:I=I+1: IF I<N THEN 280
320:GLCURSOR (0,-50)
322:LTEXT : LPRINT
325:COLOR 0: IF I=0 THEN 360
330:LPRINT "* Estimation *"
340:FOR H=0 TO I-1: LPRINT USING "&&&&&
   &&&&&&&&&&&&&";"x="+STR$ XS(H);
350:LPRINT "y="+STR$ YS(H): NEXT H
360:USING : LF 5: END
370:GLCURSOR (J,K): LPRINT "P0": RETURN
380:CONSOLE 160: LTEXT : LPRINT
390:CSIZE 2: COLOR 0: RETURN
```

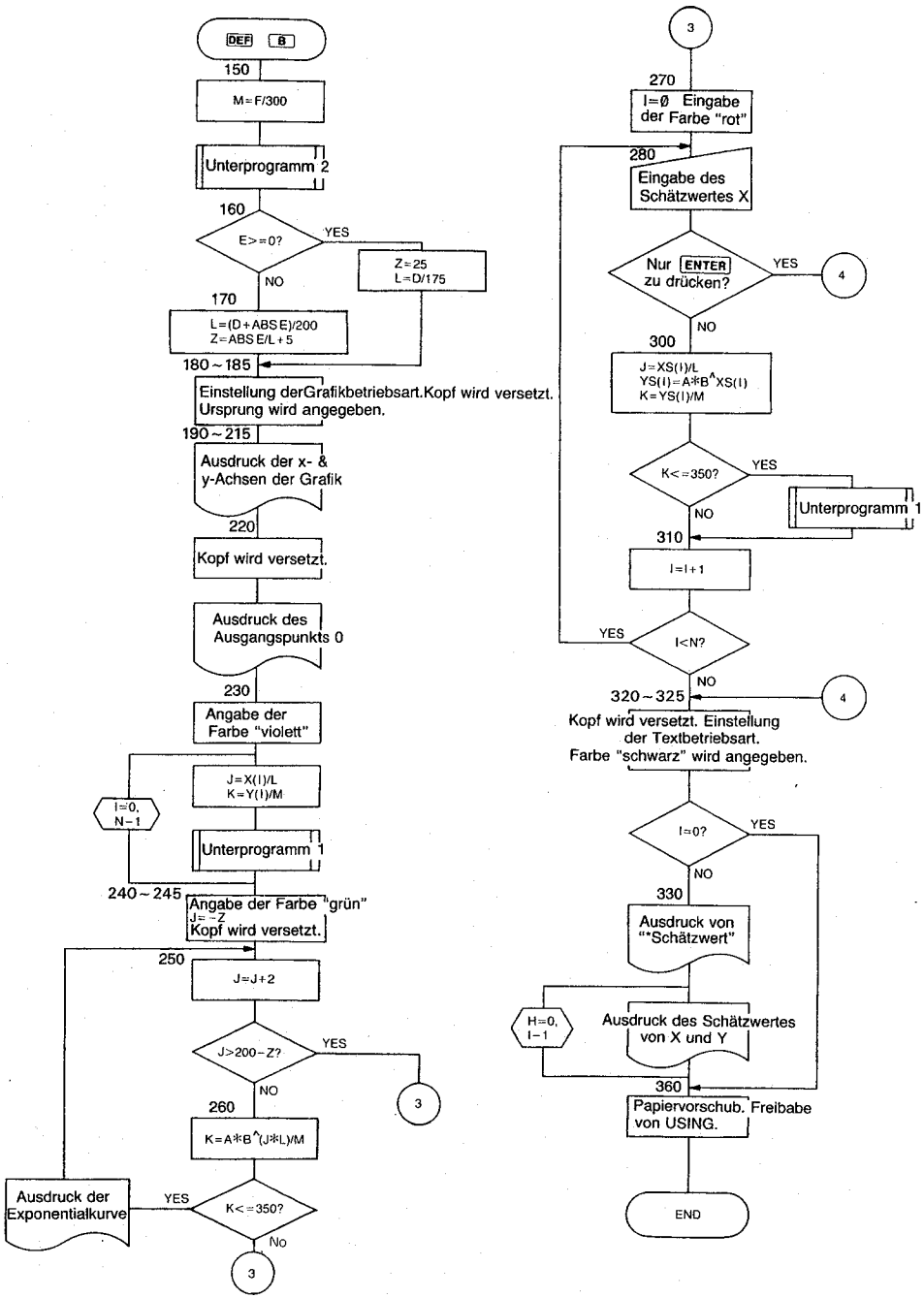
1077 Bytes

■ SPEICHERINHALT

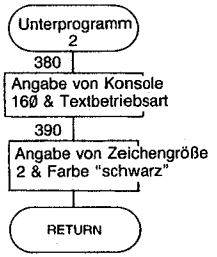
A	a
B	b', b
C	Korrelationskoeffizient r
D	X – MAX
E	X – MIN
F	Y – MAX
H	√
I	√
J	√
K	√
L	√
M	√
N	Anzahl der Daten
O	Σxi
P	ΣYi
Q	$\Sigma x^2 i$
R	$\Sigma Y^2 i$
S	$\Sigma xi \cdot Yi$
T	Sxx
U	Sxy
V	Syy
X	$\bar{x}$
Y	$lnyi$, $\bar{Y}$
Z	√
X(N–1)	X-Daten
XS(N–1)	Schätzwert für X
Y(N–1)	Y-Daten
YS(N–1)	Schätzwert für Y

■ ABLAUFDIAGRAMM




DEF B
150
M=F/300
Unterprogramm 2
160
E>=0?
YES
NO
Z=25
L=D/175
170
L=(D+ABS E)/200
Z=ABS E/L+5
180~185
Einstellung derGrafikbetriebsart.Kopf wird versetzt.
Ursprung wird angegeben.
190~215
Ausdruck der x- &
y-Achsen der Grafik
220
Kopf wird versetzt.
Ausdruck des
Ausgangspunkts 0
230
Angabe der
Farbe "violett"
J=X(I)/L
K=Y(I)/M
I=0,
N-1
Unterprogramm 1
240~245
Angabe der Farbe "grün"
J=-Z
Kopf wird versetzt.
250
J=J+2
J>200-Z?
YES
NO
3
260
K=A*B^(J*L)/M
Ausdruck der
Exponentialkurve
YES
K<=350?
No
3
3
270
I=Ø Eingabe
der Farbe "rot"
280
Eingabe des
Schätzwertes X
Nur ENTER
zu drücken?
YES
4
NO
300
J=XS(I)/L
YS(I)=A*B^XS(I)
K=YS(I)/M
K<=350?
YES
NO
Unterprogramm 1
310
I=I+1
YES
I<N?
NO
320~325
4
Kopf wird versetzt. Einstellung
der Textbetriebsart.
Farbe "schwarz" wird angegeben.
I=0?
YES
NO
330
Ausdruck von
"*Schätzwert"
H=0,
I-1
Ausdruck des Schätzwertes
von X und Y
360
Papiervorschub. Freibabe
von USING.
END

Unterprogramm
370
Kopf wird versetzt.
Aufzeichnung der Datenpunkte
RETURN


Unterprogramm 2
380
Angabe von Konsole 16Ø & Textbetriebsart
390
Angabe von Zeichengröße 2 & Farbe "schwarz"
RETURN

PROGRAMMTITEL: Vergleich von Mittelwerten
(Verbindung der Kalkulationstabellendaten mit BASIC)

Gegeben seien zwei Stichproben mit Eins-zu-Eins-Beziehung, die Stichprobenunterschiede di sollen eine Stichprobe einer Grundgesamtheit mit Normalverteilung bilden. Der Wert von t mit n-1 Freiheitsgraden kann erhalten werden von:

$$t = \frac{d}{\frac{S_d}{\sqrt{n}}}$$

Diese Formel dient zur Bestimmung, ob die beiden Stichproben-Grundgesamtheiten statistisch gleiche Mittelwerte aufweisen, dabei sind:

Mittelwerte der Unterschiede $d = \frac{\sum(x_i - y_i)}{n} = \frac{\sum x_i - \sum y_i}{n}$

Standardabweichung der Unterschiede $Sd = \left(\frac{\sum(x_i - y_i)^2 - (\sum(x_i - y_i))^2/n}{n-1}\right)^{\frac{1}{2}}$

$$= \left(\frac{\sum xi^2 - 2\sum x_i y_i + \sum yi^2 - n \cdot d^2}{n-1}\right)^{\frac{1}{2}}$$

■ Verwendung dieses Programms

1. Zuerst die Daten in der Kalkulationstabellen-Betriebsart eingeben und dann mit den Daten die statistischen Berechnungen mit zwei Variablen in der STAT-Betriebsart durchführen.

2. Dann mit den für U — Z gespeicherten statistischen Werten den Wert von t in der BASIC-Betriebsart berechnen.

Speicher	U	V	W	X	Y	Z
Statistische Werte	Σy^2	Σy	Σxy	Σx^2	Σx	n

$\bar{d}$=(Y−V)/Z→M (Mittelwert der Unterschiede)
$S\bar{d}$=SQR((X-2∗W+U−Z∗M2)/(Z−1))→RD
(Standardabweichung der Unterschiede)

t=M/SD∗SQR Z→T (Wert von t)
Df=Z−1→DF (Freiheitsgrade)

* Durch Ausführung des Programms in der BASIC-Betriebsart werden die folgenden Ergebnisse erhalten.

SD: SD = Standardabweichung der Unterschiede (Sd)
M: M = Mittelwert der Unterschiede (d)
T: T = Wert von t (t)
DF: DF = Freiheitsgrade (Df)

3. Mit der Tabelle der t-Verteilungen die Hypothese mit dem in 2. erhaltenen Wert von t überprüfen.

■ Beispiel

In einer Schule wurden zwei Klassen (mit je 10 Schülern) einem Test unterzogen, der ergab, daß die beiden Klassen das gleiche Intelligenzniveau aufweisen. Nach einem Jahr Unterricht mit unterschiedlichen Studienprogrammen A und B wurden die beiden Klassen erneut getestet, wobei die folgenden Ergebnisse erhalten wurden.

A	10	4	6	2	7	13	3	11	5	9
B	8	3	4	2	3	11	4	7	4	7

Wir wollen die folgenden Hypothesen überprüfen:

H_0: Die beiden Studienprogramme weisen die gleiche Wirksamkeit auf.

H_1: Die beiden Studienprogramme weisen mit 95% Sicherheit einen Unterschied in der Wirksamkeit auf.

Mit Freiheitsgrad 9 und 5% Signifikanzniveau ergibt sich für den kritischen Bereich der Hypothese H0 aus der Tabelle der t-Verteilungen:

$t > 2.262$

■ Bedienung

(1) Das Programm eingeben. (siehe nächste Seite)

(2) Die STAT-Betriebsart (STAT-Symbol wird angezeigt) und die Kalkulationstabellen-Betriebsart einstellen.

[STAT] [TABLE] drücken

(3) Die unten aufgeführten Bedienschritte durchführen (Dateneingabe, Vertauschung der Zeilen und Spalten der Kalkulationstabelle, Durchführung der statistischen Berechnungen mit zwei Variablen mit den Daten)

TABLE TITLE 1=?	EDUCATION [DATA]
COLUMN ITEM 1=?	1 [DATA]
COLUMN ITEM 2=?	2 [DATA]
•	
•	

Die Daten auf die gleiche Weise wie oben eingeben.

•
•

COLUMN ITEM 10=?	10	DATA
COLUMN ITEM 11=?		DATA
ROW ITEM 1=?	A	↓DATA
ROW ITEM 2=?	B	↓DATA
ROW ITEM 3=?		↓DATA
A:1=?	10	DATA
A:2=?	4	DATA

•
•

Die Daten auf die gleiche Weise eingeben.

•
•

A:10=?	9	DATA
ROW ITEM 2=B		DATA
B:1=?	8	DATA
B:2=?	3	DATA

•
•

Die Daten auf die gleiche Weise eingaben.

•
•

B:10=?	7	DATA
ROW ITEM 3=?	TABLE	
TABLE TITLE 1=EDUCATION	SHIFT	TRANSPOSE
PLEASE PUSH ENTER	ENTER	
TABLE TITLE 1=EDUCATION	►	
ROW ITEM 1=A	→X	
ROW ITEM 1 [>X]=A	►	
ROW ITEM 2=B	SHIFT	→Y
ROW ITEM 2 [>Y]=	ENTER	
STAT. CALCULATION OK?	ENTER	
0		

(4) Den Computer in die BASIC-Betriebsart schalten (RUN-Symbol wird angezeigt) und das Programm ausführen.

[BASIC] RUN [ENTER] drücken

Ergebnisse: SD = 1.567021236
M = 1.7
T = 3.43063125
DF = 9.

Da der Wert von t (T) im kritischen Bereich liegt, muß Hypothese H0 zurückgewiesen werden. Das bedeutet, daß die beiden Studienprogramme Unterschiede in der Wirksamkeit aufweisen.

Programmliste

```
10:M=(Y-V)/Z
20:SD=SQR ((X-2*W+U-Z*M
   ^2)/(Z-1))
30:T=M/SD*SQR Z
40:DF=Z-1
50:PRINT "SD=";SD
60:PRINT "M=";M
70:PRINT "T=";T
80:PRINT "DF=";DF
90:END
```

120 Bytes

Zweck der einzelnen Zeilen

Zeilennummer	Zweck
10	Berechnung des Mittelwertes
20	Berechnung der Standardabweichung
30	Berechnung von t
40	Berechnung der Freiheitsgrade
50~80	Anzeige der Ergebnisse
90	Programmende

PROGRAMMTITEL: Berechnung des Flächeninhalts eines Vielecks

Erforderliches Pheripheriegerät CE-126P

Jedes Vieleck ist theoretisch eine Zusammensetzung aus Dreiecken. Lassen Sie uns den Flächeninhalt eines Vielecks berechnen. Dieses Programm errechnet den Flächeninhalt eines Vielecks durch Unterteilung des Vielecks in Dreiecke, berechnet dann den Flächeninhalt jedes Dreiecks und schließlich die Gesamtsumme der Fläche aller vorhandenen Dreiecke.

■ BEDIENUNG

1. DEF A drücken. (Programmstart)
   Entsprechend der Abfrage auf dem Display, die Anzahl der Eckpunkte und die Koordinaten x und y jedes Eckpunktes eingeben.

2. Anschließend DEF B drücken.
   Auf dem Display werden die Nummern von drei Eckpunkten abgefragt, die einzugeben sind, wonach die Fläche des Dreiecks ausgedruckt wird. Wird vor der Eingabe der Eckpunktnummer ENTER gedrückt, wird die Gesamtfläche aller Dreiecke auf dem Drucker ausgegeben.

3. Durch Betätigen von DEF C wird die Gesamtfläche aller Dreiecke gelöscht, und das Programm ist beendet.

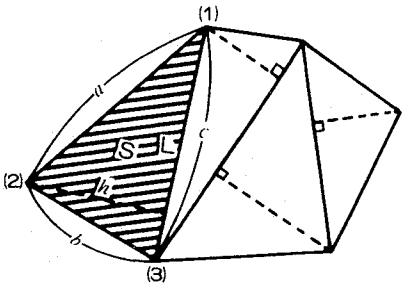
Achtung: Für die Anzahl der Eckpunkte gelten folgende Grenzen:

CE-212M (oder CE-201M) · CE-2H16M (oder CE202M) 255 Eckpunkte

■ ACHTUNG

$$S=\sqrt{s(s-a)(s-b)(s-c)}$$

$$s=\frac{a+b+c}{2}$$

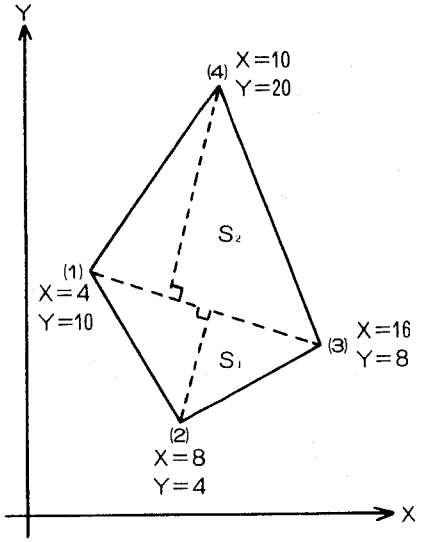


Fläche des Dreieicks $S=\frac{1}{2}hL$

mit
L: Basislänge (längste Seite unter a, b und c)
h: Höhe (Die Werte können bis zu vier Nachkommastellen eingegeben werden.)

■ BEISPIEL

Berechnen Sie die Fläche des folgenden 4seitigen Vielecks.



■ AUSDRUCK

```
Point 1x= 4
Point 1y= 10
Point 2x= 8
Point 2y= 4
Point 3x= 16
Point 3y= 8
Point 4x= 10
Point 4y= 20
  1-2-3
A=            7.211
B=            8.944
C=           12.166
H=            5.260
S=           31.996580
  1-3-4
A=           12.166
B=           13.416
C=           11.662
H=            9.838
S=           65.993304

*TS*=        97.989884
```

■ TASTENBETÄTIGUNG

<Eingabe der Koordinaten X und Y jedes Eckpunkts>

1. DEF A

```
Numbers=_
```

2. 4 ENTER

```
Point 1x=
?
```

3. 4 ENTER

```
Point 1y=
?
```

⋮

Gleiche Eingabe wie oben.

⋮

4. 20 ENTER

```
>
```

<Eingabe der Eckpunktnummern jedes Dreiecks>

1. DEF B

```
Point=_
```

2. 1 ENTER

```
Point=_
```

3. 2 ENTER

```
Point=_
```

4. 3 ENTER

```
Point=_
```

5. 1 ENTER

```
Point=_
```

6. 3 ENTER

```
Point=_
```

7. 4 ENTER

```
Point=_
```

8. ENTER

```
>
```

<Löschen der Gesamtsumme der Dreieckflächen>

1. DEF C

```
** TS CLEAR **
>
```

■ PROGRAMMLISTING

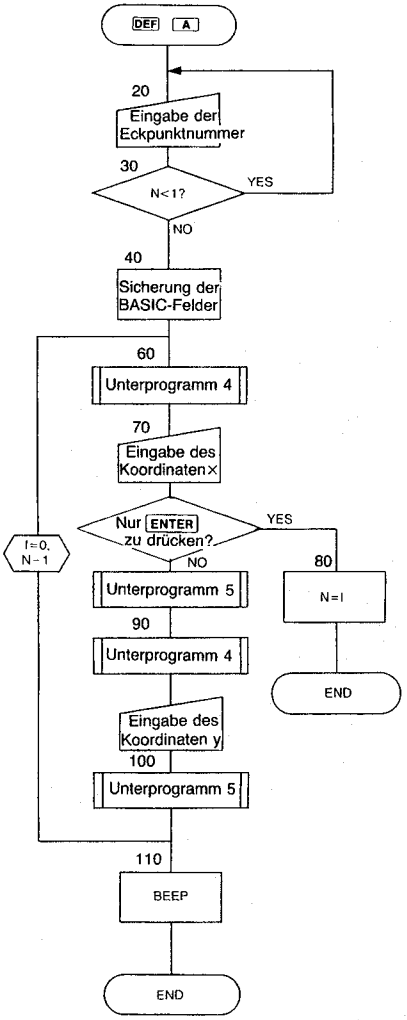
```
 10:"A":USING :CLEAR :
    WAIT 0
 20:INPUT "Numbers=";N
 30:IF N<1 BEEP 2:GOTO 2
    0
 40:DIM X(N-1),Y(N-1),B$
    (0)
 50:FOR I=0 TO N-1
 60:B$(0)="x=":GOSUB 360
 70:INPUT X(I):B$(0)="x=
     "+STR$ X(I):GOSUB 3
    70:GOTO 90
 80:N=I:END
 90:B$(0)="y=":GOSUB 360
    :INPUT Y(I)
100:B$(0)="y= "+STR$ Y(I
    ):GOSUB 370:NEXT I
110:BEEP 1:END
120:"B":USING :INPUT "Po
    int=";O,"Point=";P,"
    Point=";Q:GOTO 140
130:GOTO 310
140:IF (O<1)+(O>N)+(P<1)
    +(P>N)+(Q<1)+(Q>N)<>
    0 GOTO 120
150:C=X(O-1):D=Y(O-1):E=
    X(P-1):F=Y(P-1):G=X(
    Q-1):H=Y(Q-1)
160:X=E-C:Y=F-D:GOSUB 33
    0
170:A=X:X=G-E:Y=H-F:
    GOSUB 330
180:B=X:X=C-G:Y=D-H:
    GOSUB 330
190:C=X:IF A>X LET X=A
200:IF B>X LET X=B
210:I=(A+B+C)/2:S=SQR (I
    *(I-A)*(I-B)*(I-C))
220:J= INT ((2*S/X)*10^3
    )/10^3:L=X:GOSUB 340
230:X=L:S=X*J/2:K=K+S:L=
    A:GOSUB 340
240:A=L:L=B:GOSUB 340
250:B=L:L=C:GOSUB 340
260:C=L:L=S:GOSUB 350
270:S=L:L=K:GOSUB 350
280:K=L:LPRINT "  ";STR$
    O+"-"+STR$ P+"-"+
    STR$ Q
290:LPRINT "A= "; USING
    "##############.###";A
300:LPRINT "B= ";B:
    LPRINT "C= ";C:
    LPRINT "H= ";J:
    LPRINT USING "######
    ######.######";"S= "
    ;S:GOTO 120
310:LPRINT "":LPRINT "*T
    S*="; USING "########
    ##.######";K:END
320:"C":K=0:USING :PAUSE
    " ** TS CLEAR **":
    END
330:X=SQR (X*X+Y*Y):
    RETURN
340:L= INT (L*1000+.5)/1
    000:RETURN
350:L= INT (L*1000000)/1
    000000:RETURN
360:PAUSE "Point ";STR$
    (I+1);B$(0):RETURN
370:LPRINT "Point ";STR$
    (I+1);B$(0):RETURN
```

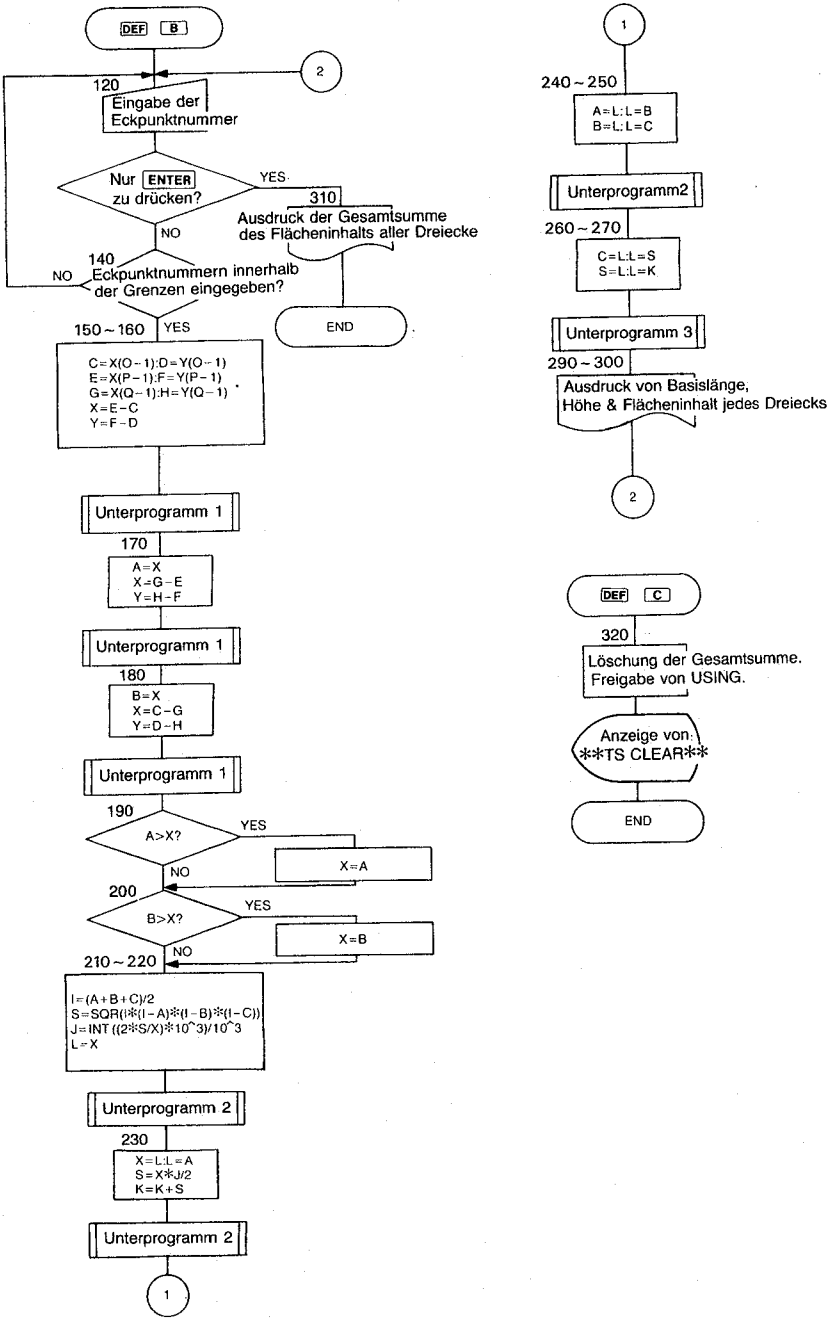
962 Bytes

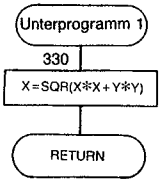
■ SPEICHERINHALT

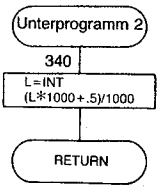
A	a
B	b
C	x_1, $\sqrt{}$
D	y_1
E	x_2
F	y_2
G	x_3
H	y_3
I	S
J	h
K	Σs
L	$\sqrt{}$
N	Anzahl der Eckpunkte
O	$\sqrt{}$
P	$\sqrt{}$
Q	$\sqrt{}$
S	S
X	$\sqrt{}$
Y	$\sqrt{}$
X(N−1)	x-Koordinaten
Y(N−1)	y-Koordinaten
B$(Ø)	$\sqrt{}$

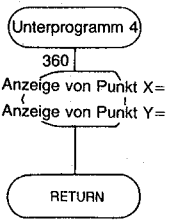
■ ABLAUFDIAGRAMM

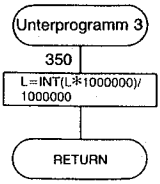


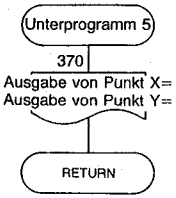

DEF B
2
120
Eingabe der Eckpunktnummer
Nur ENTER zu drücken?
YES
310
Ausdruck der Gesamtsumme des Flächeninhalts aller Dreiecke
NO
140
NO
Eckpunktnummern innerhalb der Grenzen eingegeben?
END
150 ~ 160
YES
C=X(O-1):D=Y(O-1)
E=X(P-1):F=Y(P-1)
G=X(Q-1):H=Y(Q-1)
X=E-C
Y=F-D
Unterprogramm 1
170
A=X
X=G-E
Y=H-F
Unterprogramm 1
180
B=X
X=C-G
Y=D-H
Unterprogramm 1
190
A>X?
YES
X=A
NO
200
B>X?
YES
X=B
NO
210 ~ 220
I=(A+B+C)/2
S=SQR(I*(I-A)*(I-B)*(I-C))
J=INT ((2*S/X)*10^3)/10^3
L=X
Unterprogramm 2
230
X=L:L=A
S=X*J/2
K=K+S
Unterprogramm 2
1
1
240 ~ 250
A=L:L=B
B=L:L=C
Unterprogramm2
260 ~ 270
C=L:L=S
S=L:L=K
Unterprogramm 3
290 ~ 300
Ausdruck von Basislänge, Höhe & Flächeninhalt jedes Dreiecks
2
DEF C
320
Löschung der Gesamtsumme. Freigabe von USING.
Anzeige von: **TS CLEAR**
END


Unterprogramm 1
330
X=SQR(X*X+Y*Y)
RETURN


Unterprogramm 2
340
L=INT
(L*1000+.5)/1000
RETURN


Unterprogramm 4
360
Anzeige von Punkt X=
Anzeige von Punkt Y=
RETURN


Unterprogramm 3
350
L=INT(L*1000000)/
1000000
RETURN


Unterprogramm 5
370
Ausgabe von Punkt X=
Ausgabe von Punkt Y=
RETURN

PROGRAMMTITEL: Erstellung einer Kurvengrafik
(Verbindung von Kalkulationstabellen-Daten mit BASIC)

Erforderliches Peripheriegerät: CE-140P

Es ist erstrebenswert, die in der Kalkulationstabellen-Betriebsart eingegebenen Daten auch grafisch darstellen zu können, weil die Signifikanz einzelner Daten aus einer Kurve viel deutlicher zu ersehen ist.

Wir wollen jetzt ein Programm vorstellen, das eine Kurvengrafik erstellt und sie auf dem CE-140P ausdruckt.

■ VERWENDUNG DIESES PROGRAMMS

1. Das BASIC-Programm eingeben. (siehe Seite 315)
2. Den Computer auf die Kalkulationstabellen- Betriebsart einstellen und die Daten eingeben.
3. In die RUN-Betriebsart schalten, das Programm durch Drücken von [DEF] [ENTER] starten und die Daten für die folgenden Posten eingeben.
   ① Die Anzahl der Spalten der in Schritt 2 erstellten Kalkulationstabelle spezifizieren.
   ② Den Bereich der Daten, die grafisch dargestellt werden sollen, mit der Anzahl Zeilen und Spalten der in Schritt 2 erstellten Kalkulationstabelle spezifizieren.
   * Der zulässige Bereich ist bis zu 10 Zeilen und 9 Spalten.
   ③ Wahl, ob die Datentabelle gedruckt werden soll.
   Für Drucken [Y] [ENTER] eingeben.
   Für Nichtdruck [N] [ENTER] eingeben
   ④ Zuletzt die Namen der waagerechten und senkrechten Achse mit jeweils bis zu 10 Zeichen eingeben.

4. Nach Eingabe der obigen Daten werden die Datentabelle und die Kurvengrafik oder nur die Kurvengrafik gedruckt, danach endet das Programm.

■ REFERENZ

Berechnungsmethode für den Maximalwert der Skala
Mit YX als Maximalwert der Daten
natürlicher Logarithmus: $L=INT\ (\log\ YX)$

Maximalwert der vorübergehenden Skala: $B=\frac{YX}{10^{L-1}}$

Maximalwert der Skala: $MX=I$

$$INT\ (\frac{YX}{ME}+0.9999)\times ME$$

Die Einheiten der Skala werden in den Zeilen 640 bis 680 und der Maximalwert MX der Skala wird in Zeile 700 berechnet.

■ **BEISPIEL**

Erstellung einer Kurvengrafik mit den in der Tabelle gegebenen Daten.

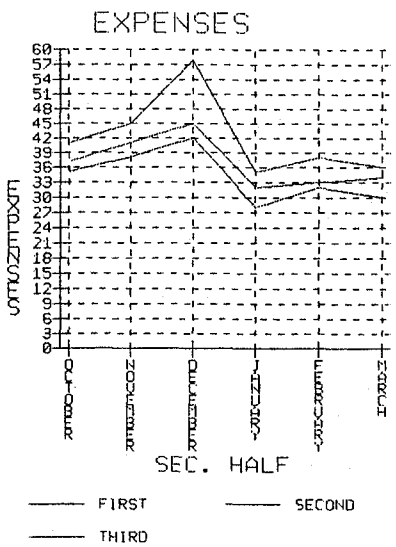
	Erster	Zweiter	Dritter
Oktober	41	35	37
November	45	38	41
Dezember	58	42	45
Januar	35	28	32
Februar	38	32	33
März	36	30	34

(Einheit: Zehntausend Dollar)

Titel: Ausgaben
Titel der waagerechten Achse: 2. Hälfte
Titel der senkrechten Achse: Ausgaben

■ **AUSDRUCK** *(Ausdruck in Farbe. Siehe Seite I)*

	FIRST	SECOND	THIRD
OCTOBER	41	35	37
NOVEMBER	45	38	41
DECEMBER	58	42	45
JANUARY	35	28	32
FEBRUARY	38	32	33
MARCH	36	30	34



■ TESTENBEDIENUNG

Die Schritte ① bis ⑪ umfassen die Dateneingabe in der STAT- und Kalkulationstabellen-Betriebsart.

① [TABLE]	TABLE TITLE 1=?
② EXPENSES [DATA]	COLUMN ITEM 1=?
③ FIRST [DATA]	COLUMN ITEM 2=?
•	
•	
Die Daten auf die gleiche Weise eingeben.	
•	
•	
④ THIRD [DATA]	COLUMN ITEM 4=?
⑤ [DATA]	ROW ITEM 1=?
⑥ OCTOBER [↓DATA]	ROW ITEM 2=?
•	
•	
Die Daten auf die gleiche Weise eingeben.	
•	
•	
⑦ MARCH [↓DATA]	ROW ITEM 7=?
⑧ [↓DATA]	OCTOBER:FIRST=?
⑨ 41 [DATA]	OCTOBER:SECOND=?
⑩ 35 [DATA]	OCTOBER:THIRD=?
•	
•	
Die Daten auf die gleiche Weise eingeben.	
•	
•	
⑪ 34 [DATA]	ROW ITEM 7=?
⑫ [BASIC]	>
⑬ [DEF] [A]	NO. OF TABLE COLUMN=

⑭ 3 [ENTER]	RANGE OF GRAPH
	COLUMN ITEM=
⑮ :3 [ENTER]	ROW ITEM=
⑯ 1:6 [ENTER]	PRINT TABLE DATA (Y/N)
⑰ Y [ENTER]	X-AXIS TITLE=
⑱ SEC. HALF [ENTER]	Y-AXIS TITLE=
⑲ EXPENSES [ENTER]	>

■ PROGRAMMLISTING

```
 10:"A": WAIT : USING : CLEAR
 20:INPUT "NO. OF TABLE COLUMN=";A
 30:PAUSE "RANGE OF GRAPH"
 40:INPUT "COLUMN ITEM=";C$: GOSUB 1500
    : IF G>0 LET D=G:F=H: GOTO 60
 50:BEEP 1: GOTO 40
 60:INPUT "ROW ITEM=";C$: GOSUB 1500:
    IF G>0 THEN 80
 70:BEEP 1: GOTO 60
 80:XK=F-D+1:F=F-1:D=D-1
 90:YK=H-G+1:H=H-1:G=G-1
100:DIM X$(XK),HA(XK),Y$(YK),DD(XK,YK),
    T$(3)*22
110:INPUT "PRINT TABLE DATA (Y/N)";C$
120:IF (C$<>"Y") AND (C$<>"N")=1 BEEP 1
    : GOTO 110
130:INPUT "X-AXIS TITLE=";T$(1)
140:IF LEN T$(1)>10 BEEP 1: GOTO 130
150:INPUT "Y-AXIS TITLE=";T$(2)
160:IF LEN T$(2)>10 BEEP 1: GOTO 150
170:FOR I=D TO F: FOR J=G TO H:DD(I-D+1
    ,J-G+1)=D0(J,I): NEXT J: NEXT I
180:T$(0)=T0$(A)
190:FOR I=D TO F:X$(I-D+1)=T0$(I): NEXT
    I
200:FOR J=G TO H:Y$(J-G+1)=D0$(J): NEXT
    J
210:SP$="        ":E$=CHR$ 27
220:LPRINT E$+"@": LPRINT E$+"b"
230:IF C$="N" THEN 500
240:S1= INT (XK/4)+1:X=0: FOR O=1 TO S1
250:S2=3: IF O=S1 LET S2=XK-4*O+4
260:GLCURSOR (0,-20)
270:SORGN :X2=108*(S2+1)
280:LLINE -(X2,0)
290:FOR I=0 TO S2+1
300:GLCURSOR (X2-I*108,0)
310:RLINE -(0,-20): NEXT I
320:FOR I=0 TO S2
330:GLCURSOR (12+I*108,-16)
340:LPRINT "P";X$(X+I): NEXT I
350:FOR Y=1 TO YK
360:GLCURSOR (0,-20)
370:SORGN
380:RLINE -(X2,0)
390:FOR I=0 TO S2+1
400:GLCURSOR (X2-I*108,0)
410:LLINE -(X2-I*108,-20): NEXT I
420:FOR I=0 TO S2
430:GLCURSOR (108*I-2,-16)
440:IF X+I=0 LPRINT "P ";Y$(Y): GOTO 46
    0
450:LPRINT "P"; USING "##########";DD(X+
    I,Y)
460:NEXT I: NEXT Y
470:GLCURSOR (X2,-20)
480:LLINE -(0,-20)
490:SORGN :X=X+4: NEXT O
500:GLCURSOR (0,-50)
510:SORGN
520:USING :WX=480:ZX=500:ZY=360
530:YN=100000000:YX=-YN
540:FOR Y=1 TO YK: FOR X=1 TO XK:D=DD(X
    ,Y)
550:IF D>YX LET YX=D
560:IF D<YN LET YN=D
570:NEXT X: NEXT Y
580:PY=-40-ZY
590:MS=(LEN STR$ YX)*12: IF MS<12 LET M
    S=36
600:IF ZX+MS+60>WX LET ZX=WX-MS-60
610:PX=WX-ZX-5
620:KY=YK-1:XH=(ZX-20)/KY
630:L= INT LOG YX:A=10^(L-1):B=YX/A
640:IF B<=12 LET P=1: GOTO 680
650:IF B<=20 LET P=2: GOTO 680
660:IF B<=60 LET P=3: GOTO 680
670:P=10
680:ME=P*A
690:MN=0: IF YN<0 LET MN= INT (YN/ME)*M
    E
700:MX= INT (YX/ME+.9999)*ME
710:YH= INT (ZY/(ABS (MX-MN)/ME))
720:YA=ME:PA=YH/YA
730:GLCURSOR (120,-20)
740:CSIZE 4
750:LPRINT "P";T$(0)
760:CSIZE 2
770:GLCURSOR (PX,PY)
780:SORGN
790:LLINE -(ZX,0)
800:SW=1:Y=0:X=ZX
810:RP= INT ((MX-MN-YA)/YA)
820:FOR LL=0 TO RP:K=MN+YA+LL*YA:Y=Y+YH
830:IF SW<>1 THEN 860
840:GLCURSOR (X,Y)
850:LLINE -(0,Y),6: GOTO 880
860:GLCURSOR (0,Y)
870:LLINE -(X,Y),6
880:SW=-SW
890:NEXT LL
900:GLCURSOR (0,0)
910:XM=-XH+10:YU=0:YD=-10:NY=-60
920:FOR K=1 TO YK:XM=XM+XH
930:GLCURSOR (XM,YU)
940:LLINE -(XM,YD),0
950:X=XM-4:Y=-12:T$(3)=Y$(K):ST=14
960:GOSUB 1450
970:IF Y<NY LET NY=Y
980:NEXT K
990:CSIZE 3:MS=LEN T$(1):X= INT ((ZX-MS
    *18)/2):Y=NY-25
1000:GLCURSOR (X,Y)
1010:LPRINT "P";T$(1)
1020:CSIZE 2
1030:GLCURSOR (0,0)
1040:LLINE -(0,ZY):SW=1:X=10:Y=ZY
1050:FOR K=1 TO YK: IF SW<>1 THEN 1080
1060:GLCURSOR (X,Y)
1070:LLINE -(X,0),6: GOTO 1100
1080:GLCURSOR (X,0)
1090:LLINE -(X,Y),6
1100:X=X+XH:SW=-SW: NEXT K
1110:GLCURSOR (0,0):YM=-YH:XL=-10:XR=1
     0
1120:RP= INT ((MX-MN)/YA)
1130:FOR LL=0 TO RP:K=MN+LL*YA:YM=YM+Y
     H
1140:MS=(LEN STR$ K):X=-MS*12-8:Y=YM-6
1150:GLCURSOR (X,Y)
1160:LPRINT "P";STR$ K
1170:GLCURSOR (XL,YM)
1180:LLINE -(XR,YM),0
1190:IF X<NX LET NX=X
```

```
1200:NEXT LL
1210:CSIZE 3:T$(3)=T$(2):Y= INT (ZY-(Z
     Y-MS*20)/2):X=NX-30:ST=20
1220:GOSUB 1450
1230:CSIZE 2
1240:GLCURSOR (0,0)
1250:L=0:CL=1
1260:COLOR 1
1270:FOR R=1 TO XK:X=10:Y= INT ((DD(R,
     1)-MN)*PA):HA(R)=CL*100+L
1280:GLCURSOR (X,Y)
1290:FOR G=2 TO YK:X=X+XH:Y= INT ((DD(
     R,G)-MN)*PA)
1300:LLINE -(X,Y),L: NEXT G
1310:CL=CL+1: IF CL=4 LET CL=0:L=L+1
1320:COLOR CL: NEXT R
1330:Y=NY-65:X= INT (ZX/6):SW=1
1340:FOR K=1 TO XK: IF SW<>1 THEN 1360
1350:GLCURSOR (NX-5,Y): GOTO 1370
1360:GLCURSOR (ZX/2,Y)
1370:CL= INT (HA(K)/100)
1380:COLOR CL
1390:RLINE (0,0)-(X,0),HA(K)-CL*100
1400:LPRINT "R20,-6": LPRINT "P";X$(K)
1410:IF SW=-1 THEN LET Y=Y-40
1420:SW=-SW: NEXT K
1430:GLCURSOR (-PX,Y-100)
1440:LPRINT E$+"@": END
1450:MS=LEN T$(3)
1460:FOR KK=1 TO MS:Y=Y-ST
1470:GLCURSOR (X,Y):TC$=MID$ (T$(3),KK
     ,2)
1480:LPRINT "P";LEFT$ (TC$,1)
1490:NEXT KK: RETURN
1500:G=VAL C$:E=LEN STR$ D
1510:E=E+1: IF MID$ (C$,E,1)<>":" THEN
     1510
1520:H=VAL RIGHT$ (C$,LEN C$-E): IF G>
     H LET G=-1
1530:RETURN
```

2906 Bytes

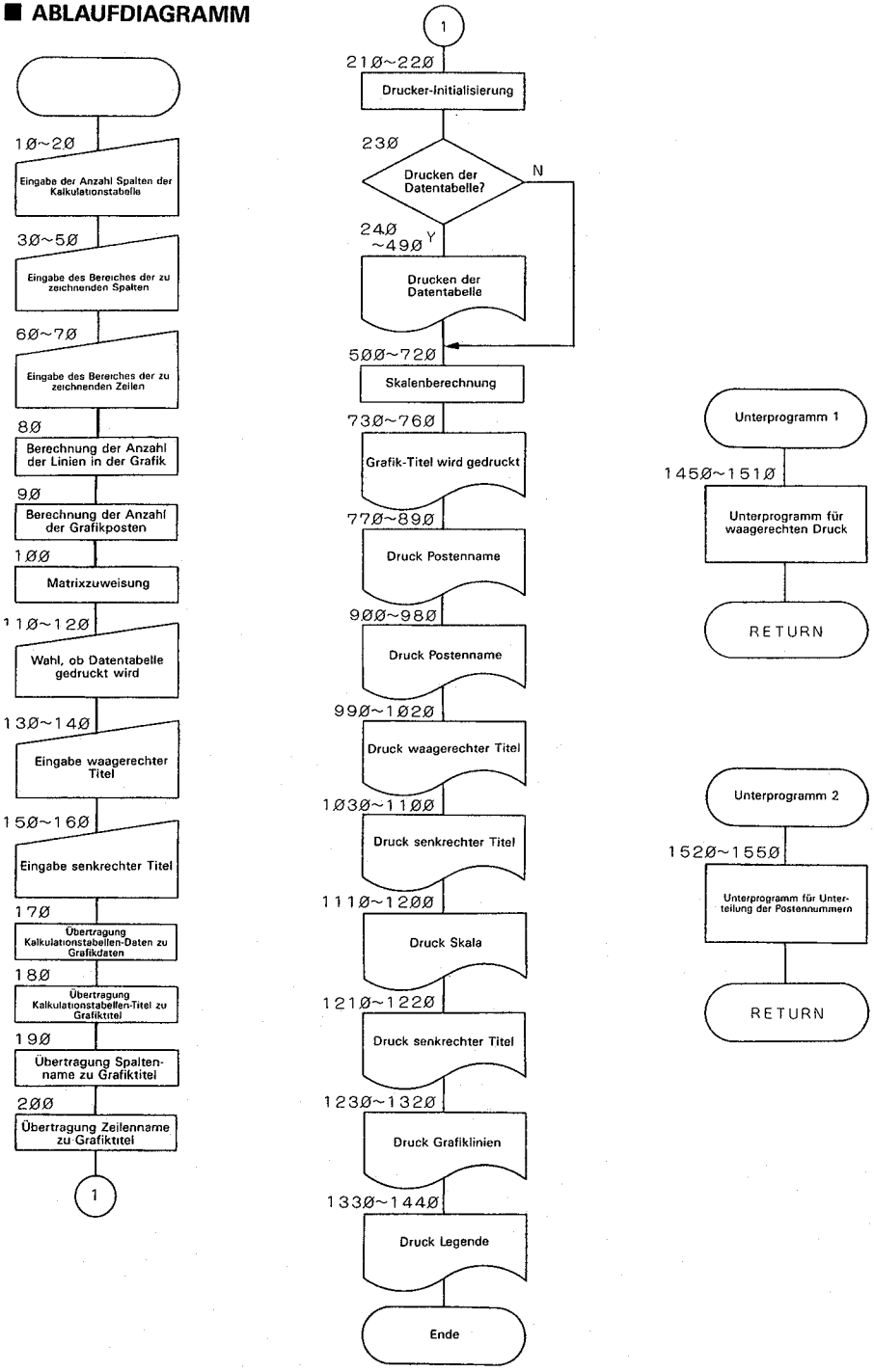
■ SPEICHERINHALT

A	✓
B	Maximalwert der vorübergehenden Skala
D	✓
F	✓
G	✓
H	✓
I	✓
K	Schleifenzähler oder Eingabe variable
L	Kurve
	Linientyp
O	Schleifenzähler
P1	✓
R	Schleifenzähler
X	Schleifenzähler oder x-Koordinaten
Y	Schleifenzähler oder y-Koordinaten
S1	✓
S2	✓
S3	✓
CL	Farbe
KK	Schleifenzähler
KY	✓
LL	Schleifenzähler
ME	Skaleneinheit
MN	✓
MS	✓
MX	Maximalwert der Skala
NX	✓
NY	✓
PA	✓

PX	Ursprung der x-Koordinaten
PY	Ursprung der y-Koordinaten
RP	Anzahl der Skalen
ST	✓
SW	Kennzeichnung
WX	Maximalwert der x-Koordinaten
XH	✓
XK	Anzahl der Kurvengrafiken
XL	✓
XM	✓
XR	Koordinate des zu druckenden Skalenwertes
YA	Abstand der Postenlinie
YD	✓
YH	Skalenabstand
YK	✓
YM	✓
YN	Minimalwert der Daten
YU	✓
YX	Maximalwert der Daten
ZX	✓
ZY	✓
DO(i,j)	Kalkulationstabellen-Daten
DD(XK,YK)	Daten
HA(XK)	Typ der Kurvengrafik
C$	✓
DO$(j)	Kalkulationstabellen-Daten
E$	ESC-Code
SP$	Leerstelle
TC$	✓
DO$(j)	Kalkulationstabellen-Daten

T0$(i)	Kalkulationstabellen-Daten
T$(3)	Titel
X$(XK)	Name der Kurvengrafik
Y$(YK)	Postenname

■ ABLAUFDIAGRAMM



PROGRAMMTITEL: Übertragung von Programmdateien

Erforderliches Peripheriegerät CE-130T

Bisher waren nur Cassettenbänder (Mikrocassetten) als externe Speichermöglichkeit für tragbare Computer erhältlich. Jetzt können jedoch auch Personalcomputer-Disketten als externer Speicher über den Pegelwandler CE-130T für den tragbaren Computer PC-1425 eingesetzt werden. Hier eine Beschreibung zur Übertragung zu und vom Personalcomputer PC-1425 mit dem PC-7000 als Beispiel eines Personalcomputers.

< VORBEREITUNG >

- Zusätzlich zum Personalcomputer PC-7000 muß ein Kabel zum Anschluß des CE-130T an die Schnittstelle des PC- 7000 vorbereitet werden. Die folgende Tabelle zeigt das für den Schnittstellenanschluß erforderliche Verbindungskabel und Stiftverbindungen.

CE-130T

Signalname	Stift-Nr.
—	1
SD	2
RD	3
RS	4
CS	5
DR	6
SG	7
CD	8
RR	11
ER	20

Männlicher Stecker
DB-25P oder äquivalentes Teil

PC-7000

Stift-Nr.	Signalname
1	FG
2	SD
3	RD
4	RS
5	CS
6	DR
7	SG
8	CD
20	ER

Weiblicher Stecker
DB-25S oder äquivalentes Teil

..... gibt an, daß ein Anschluß zwischen diesen beiden Stiften nicht unbedingt erforderlich ist.

★ ÜBERTRAGUNG DER PROGRAMMDATEI (PC-1425→ PC-7000)

Zunächst eine Beschreibung der Übertragung einer Programmdatei vom PC-1425 zum PC-7000. Der PC-7000 speichert die vom PC-1425 empfangenen Daten auf einer 5-¼ Zoll-Diskette. Der empfangene Inhalt der Programmdatei kann auf dem Schirm des PC-7000 abgerufen und überprüft werden.

■ BEDIENUNG

PC-1425

<Programmeingabe>

① Ein Programm eingeben, daß zum PC-7000 übertragen werden soll.

<Vorbereitung zur Übertragung der Programmdatei>

② Betriebsart PRO oder RUN einstellen.

③ CLOSE [ENTER]

④ OPEN "1200, N, 8, 1, A, L, & 1A" [ENTER]
→ SAVE-Befehl kann jetzt ausgeführt werden.

<Übertragung der Programmdatei>

⑤ SAVE [ENTER]
→ Der Computer überträgt die Programmdatei.

PC-7000

< Programmeingabe>

① Programm 1-1 eingeben.

<Empfang der Programmdatei>

② RUN [↵]
→ Der Computer führt Programm 1-1 aus.
→ Der Computer wartet auf den Empfang der Programmdatei.

③ Der Computer empfängt die Programmdatei.
→Der Inhalt der empfangenen Datei wird Zeile für Zeile auf dem Schirm angezeigt.

④ Dateinamen des empfangenen Programms eingeben.
→Die Programmdatei ist jetzt auf der Diskette gespeichert.

■ PROGRAMMLISTUNG 1-1

```
10 CLEAR:DIM A$(1000)
20 CLOSE:OPEN "COM1:1200,N,8,1" AS #1
30 Z$=INPUT$(1,1):IF Z$=CHR$(&H1A) THEN 100
40 LINE INPUT #1,A$
50 A$=Z$+A$
60 PRINT A$
70 A$(I)=A$
80 I=I+1
90 GOTO 30
100 INPUT "FILE NAME=";FI$
110 OPEN "O",#2,FI$
120 FOR J=0 TO I-1
130 PRINT #2,A$(J)
140 NEXT J
150 CLOSE
160 END
```

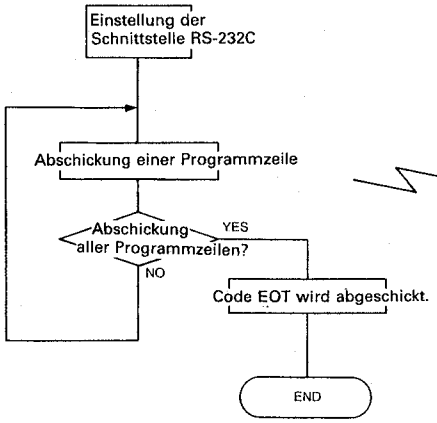
Hinsichtlich des PC-1425 sind keine Eingaben zur Übertragung erforderlich.

■ SPEICHERINHALT

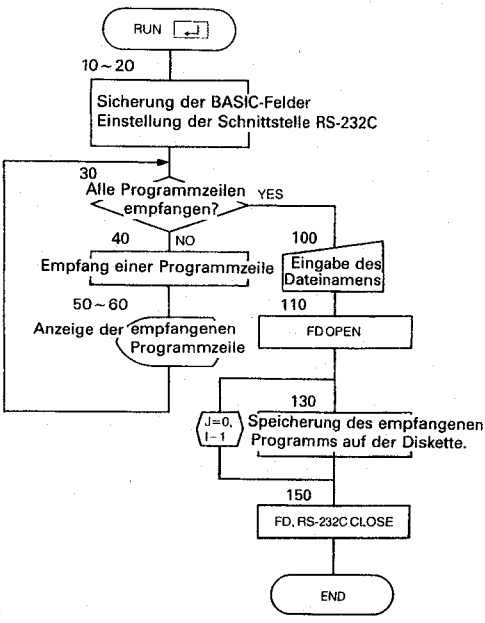
I	Zähler
F	Schleifenzähler
A$	Zur Eingabe der Programmzeile
Z$	√
FI$	Dateiname
A$(1000)	√

■ ABLAUFDIAGRAMM

PC-1425(Manuelle Einstellung)



PC-7000



• ÜBERTRAGUNG DER PROGRAMMDATEI (PC-7000 →PC-1425)

Hier eine Beschreibung der Übertragung der auf der Diskette des PC-7000 gespeicherten Programmdatei auf den PC-1425. Der zu übertragende Inhalt der Programmdatei kann auf dem Schirm des PC-7000 abgerufen und überprüft werden.

■ BEDIENUNG

PC-7000

<Programmeingabe>

① Programm 2-1 eingeben.

<Übertragung der Programmdatei>

② RUN ↵
→ Der Computer führt Programm 2-1 aus.
③ Den Namen der zu übertragenden Programmdatei eingeben.
→ Der Inhalt der empfangenen Datei wird Zeile für Zeile auf dem Schirm angezeigt.
→ Der Computer überträgt die Programmdatei.

PC-1425

<Vorbereitung für den Empfang der Programmdatei>

① Betriebsart PRO oder RUN einstellen.
② CLOSE [ENTER]
③ OPEN "1200, N, 8, 1, A, L, & 1A" [ENTER]
→ LOAD-Befehl kann ausgeführt werden.

<Empfang der Programmdatei>

④ LOAD [ENTER]
→ Der Computer wartet auf den Empfang der Programmdatei.
Der Computer empfängt die Programmdatei.
→ Übertragung beendet.

■ PROGRAMMLISTUNG 2-1

```
10 CLOSE:OPEN "COM1:1200,N,8,1" AS #1
20 INPUT "FILE NAME=";FI$
30 OPEN "I",#2,FI$
40 IF EOF(2)=-1 THEN PRINT #1,CHR$(&H1A)::CLOSE:END
50 LINE INPUT #2,A$
60 PRINT A$
70 PRINT #1,A$+CHR$(&HD):
80 GOTO 40
```

Hinsichtlich des PC-1425 sind keine Eingaben zur Übertragung erforderlich.

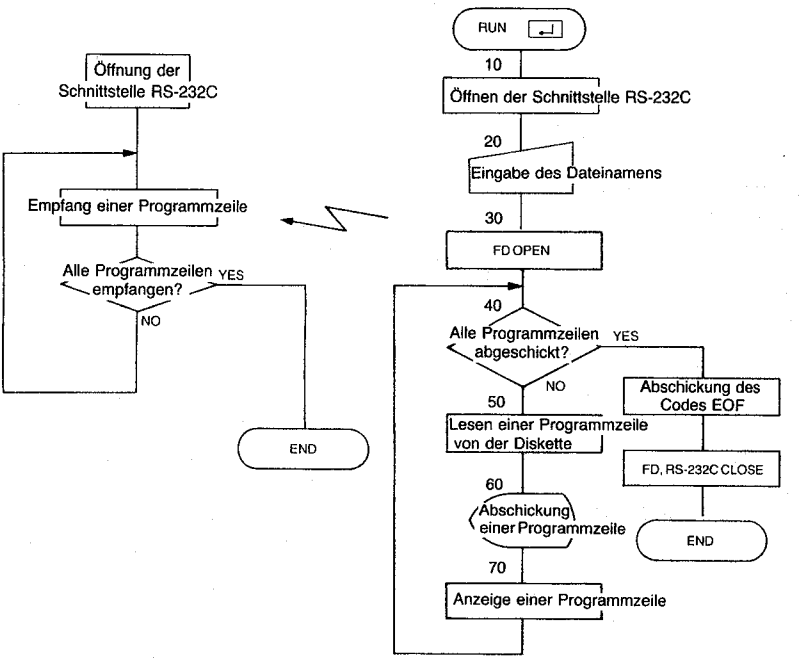
■ SPEICHERINHALT

A$	Zur Ausgabe einer Programmzeile
FI$	Dateiname

■ ABLAUFDIAGRAMM

PC-1425 (Manuelle Einstellung)

PC-7000



PROGRAMMTITEL: Zahlenspiel

In diesem Programm werden dreistellige Zufallszahlen erzeugt, die erraten werden müssen. Statt anstrengender Programme und Studien für die nächste Prüfung, versuchen Sie es zur Abwechslung einmal hiermit. Es geht darum mit möglichst wenigen Versuchen einen Treffer zu erzielen.

■ BEDIENUNG

1. [DEF] [A] (Programmstart)
2. Auf dem Display wird "X=" angezeigt. Geben Sie jetzt eine dreistellige Zahl ein, von der Sie meinen, daß sie der Computer erzeugt haben könnte. Auf dem Display wird dann die Anzahl der bisherigen Versuche und die von Ihnen eingegebene dreistellige Zahl zusammen mit einem Kommentar angezeigt (etwa 1 Sekunde später).

   Beispiel:

   - Kommentar: 1 1
     Erscheint obige Anzeige, bedeutet die erste Stelle (1) nach dem Kommentar, daß eine Stelle der von Ihnen eingegebenen Zahl der erzeugten Zufallszahl hinsichtlich von Position und Wert entspricht. Die zweite Stelle (1) zeigt, daß eine weitere Stelle der eingegebenen Zahl hinsichtlich des Werts richtig ist, aber die Position dieser Stelle falsch ist.

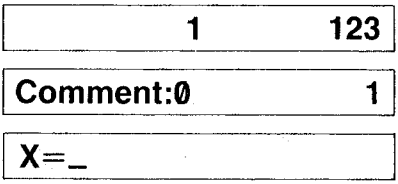
   - Kommentar: 3 0
     Diese Anzeige bedeutet, daß die von Ihnen eingegebene Zahl hinsichtlich der Stellen als auch der Position der Stellen richtig ist. Nach einem Treffer wie diesem erscheint auf dem Display die Anzeige "VERY GOOD!" sowie die Anzahl der bisherigen Versuche.

Achtung: Es können nur dreistellige Zahlen eingegeben werden.

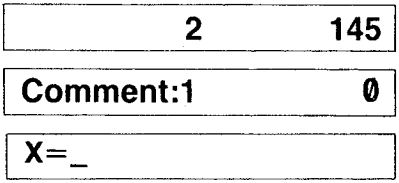
■ TASTENBETÄTIGUNG

1. DEF A

2. 123 ENTER



3. 145 ENTER

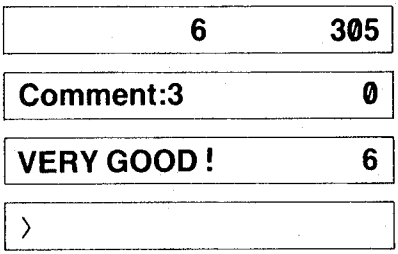


⋮

Gleiche Eingabe wie oben.

⋮

4. 305 ENTER



■ PROGRAMMLISTING

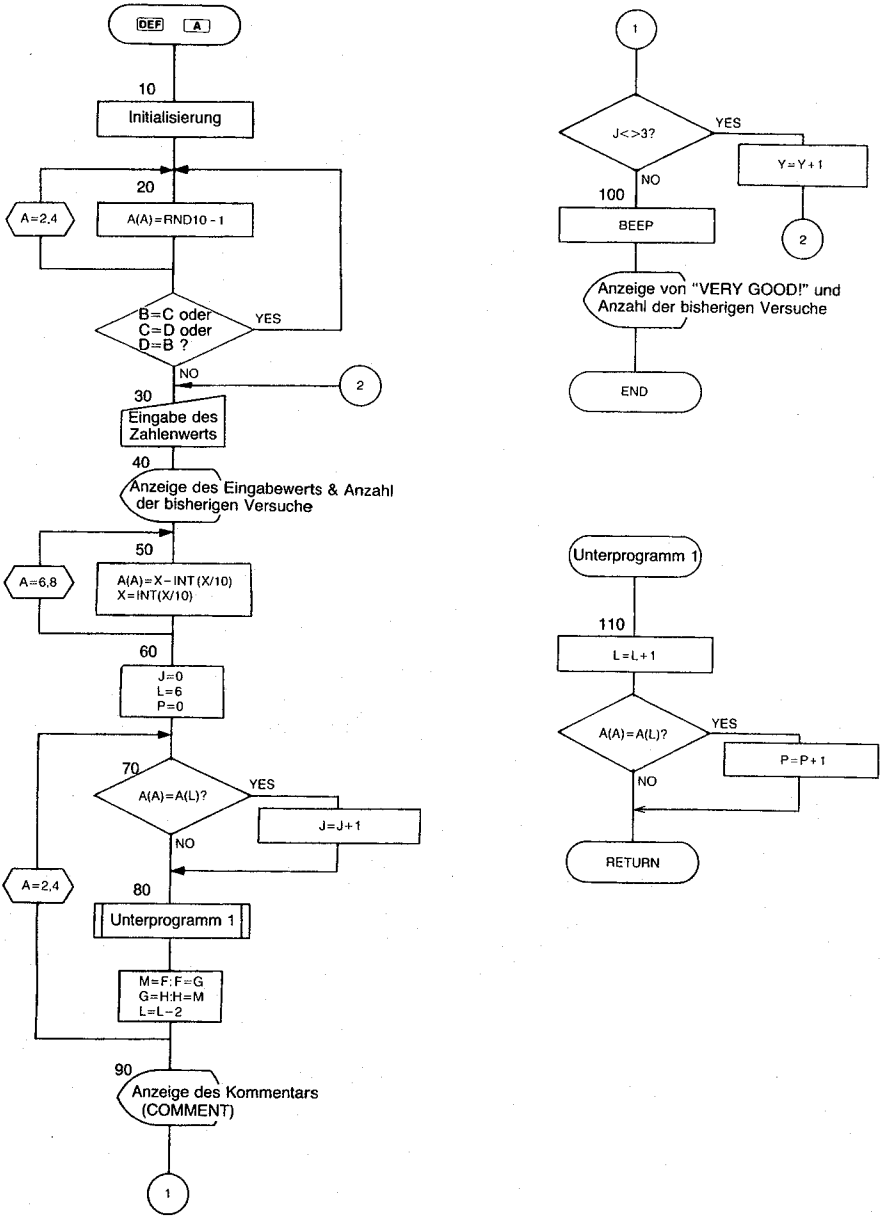
```
10:"A":CLEAR :RANDOM :Y
   =1
20:FOR A=2 TO 4:A(A)=
   RND 10-1:NEXT A:IF (
   B=C)+(C=D)+(D=B)<>0
   THEN 20
30:BEEP 1:INPUT "X=";X
40:USING :PAUSE Y,X
50:FOR A=6 TO 8:A(A)=X-
   INT (X/10)*10:X= INT
   (X/10):NEXT A
60:J=0:L=6:P=0
70:FOR A=2 TO 4:IF A(A)
   =A(L) LET J=J+1
80:GOSUB 110:GOSUB 110:
   M=F:F=G:G=H:H=M:L=L-
   2:NEXT A
90:PAUSE "Comment:";
   USING "####";J;P:IF
   J<>3 LET Y=Y+1:GOTO
   30
100:BEEP 2:PRINT "VERY G
   OOD ! ";Y:END
110:L=L+1:IF A(A)=A(L)
   LET P=P+1
120:RETURN
```

309 Bytes

■ SPEICHERINHALT

A	√
B	√
C	3-stellige Zahl
D	√
F	√
G	√
H	√
J	Kommentar
L	√
M	√
P	Kommentar
X	Eingabewert
Y	Anzahl der Versuche

■ ABLAUFDIAGRAMM



ANHANG I. INDEX

ABS 223
ACS 223
Addition 63
AND 88
Anzeige 11
AREAD 153
ASC 228
ASN 223
ATN 224
Auf-/Abrundung 61
Ausschalten 24
BASIC 248
BDS 215
BEEP 155
Betriebshinweise 14
Betriebstart 12
C.CE Key 61
CE-515P 120
CHAIN 156
CHR$ 228
CLEAR 158
CLOAD 136
CLOAD? 137
CLOSE 231
COLOR 201
CONSOLE 232
CONT 138
COS 224
CROTATE 203
CSAVE 139
CSIZE 204
CUR 224
DATA 159
Datenspeicherung 117
Definable Keys 10, 269
DEGREE/GRAD/RADIAN 160, 169, 189
DELETE 140
DIM 161
Division 64
Drucker 113
END 163

Editierfunktionen 34, 76
Einfache Variable 93
Einschalten 23
ERASE 164
EXP 224
Exponentialfunktionen 70
FACT 224
Fakultät 70
FDS 215
Fehlende Daten 30
Fehlermeldung/Fehlersuche 109
Feldvariable 96
FOR...TO...STEP 165
GLCURSOR 205
GOSUB 167
GOTO 141, 168
GRAPH 206
Hexadezimalzahlen 78
IF...THEN 170
Indizierte Variable 93
INKEY$ 220
INPUT 171
INPUT# 173
INPUT#1 233
INT 225
Kassetten-Interface 116
LEFT$ 229
LEN 229
LET 176
LF 208
LIST 142
LLINE 209
LLIST 143, 235
LN 225
LOAD 237
LOG 225
Logarithmische Funktionen 70
Löschen der Eingabe oder einer Fehlermeldung 61
Löschen einer Zeile 108
LPRINT 177, 178, 239
LTEXT 212
MAX 38
MEM 221

MERGE....145
MID$....229
MIN....38
Multiplikation....64
NEW....148
NEXT....180
Nachkommastellen....61
NCR....225
NDS....216
NOT....89
NPR....225
Numerische Feldvariable....97
ON...GOSUB....181
ON...GOTO....182
OPEN....242
OPEN$....244
OR....88
PASS....149
PAUSE....183
PDS....216
PFD....217
PI....222
PND....217
Potenzfunktion....65
PRINT....185
PRINT#....187
PRINT#....245
Programmaufbau....105
Programmausführung....108
Programmeingabe....105
Programmerstellung....104
Prozent....35
PTD....218
PXD....218
RANDOM....190
RCP....226
READ....191
Rechenbereich....273
Rechengenauigkeit....74
REM....192
RENUM....150
RESTORE....193
RETURN....194

RIGHT$ 229
RLINE 213
RND 226
ROT 226
RUN/GOTO 152
SAVE 247
Schlüsselwörter, vorprogrammierte 76
SGN 226
SIN 227
SORT 39
SORGN 214
SQR 227
SQU 227
Speicherrechnung 65
SRCH 40
Statistche Berechnungen 43
STOP 195
STR$ 229
Stromversorgung 16
Substraktion 63
Summe/Mittelwert 36
TABLE-Taste 26
TAN 227
Tastaturabdeckung 8
Tastenfunktionen 268
TDS 219
TEN 227
TEXT 249
Textausdrucke 85
Textfunktionen 248
Textfeldvariable 98
Transponieren 35
Trigonometrische Funktionen und Umkehrfunktion 69
TROFF 196
TRON 197
USING 198
VAL 230
Variable A 94
Variablen 91
Vergleichsausdrücke, logische 86
Verteilungsfunktionen 49, 215
WAIT 200
Wissenschaftliche Schreibweise 62, 75

Wurzelfunktion 65
XDS.....219